AFRL-IF-RS-TR-2003-74
Final Technical Report
April 2003

GLOBAL INFORMATION ENTERPRISE SIMULATION LABORATORY (GIESIM LAB)

Prediction Systems, Incorporated

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2003-74 has been reviewed and is approved for publication.

APPROVED:

SIAMAK TABRIZI
Project Engineer

FOR THE DIRECTOR:

WARREN H. DEBANY, Technical Advisor
Information Grid Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	APRIL 2003	Final May 02 – Jan 03

4. TITLE AND SUBTITLE
GLOBAL INFORMATION ENTERPRISE SIMULATION LABORATORY (GIESIM LAB)

5. FUNDING NUMBERS
C - F30602-99-D-0001/ TASK 14
PE - 62702F
PR - 4519
TA - JM
WU - 14

6. AUTHOR(S)
William C. Cave and John H. Fikus

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Prediction Systems, Incorporated
309 Morris Avenue
Spring Lake New Jersey 07762

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory/IFGC
525 Brooks Road
Rome New York 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2003-74

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Siamak Tabrizi/IFGC/(315) 330-4823/ Siamak.Tabrizi@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 Words)***
Most large-scale force-level simulations assume perfect communications. The goal of the Global Information Enterprise Simulation (GIESim) Lab is the development of a modeling and simulation framework that bridges these communication gaps. PSI was chosen to participate in the formation of the GIESim Lab. PSI performed several analyses on the requirements and architecture needed to ensure success of the GIESim Lab. Analyses performed by PSI spanned user and application requirements, requirements for generic infrastructure, and a generic approach to model architecture and simulation development needed to cut time and cost of producing valid realizations that met performance needs. To elucidate on the capabilities of General Simulation System (GSS) created by PSI, PSI prepared an in-depth overview of GSS attributes applicable to GIESim. Also, PSI created a model taxonomy to help frame the GIESim models selection process. In addition, PSI demonstrated a multi-simulation environment using HLA and TCP/IP as a proof of concept for the GIESim approach. This multi-simulation demonstration consisted of separate simulations of Emitters, Unattended Ground Sensors, an Integrated Air Defense System (LADS), and an Operations Management System (OMS) for the NFSS Project. PSI suggested several research topics for FY03/04 to ensure the success of the GIESim Lab.

14. SUBJECT TERMS
GIESim, Multi-Simulation Environment, Modeling & Simulation, DTIG, General Simulation System, GSS, Netted Full Spectrum Sensors, NFSS, Communication Systems, Integrated Air Defense System, IADS

15. NUMBER OF PAGES
82

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

TABLE OF CONTENTS

List of Figures and Tables

List of Tables

1. BACKGROUND

As more sophisticated systems are required to combat rapidly growing threats, the Air Force faces greater challenges in getting these systems from an R&D concept to production. A significant challenge is dealing with the level of design complexity imposed by these systems. As software replaces hardware to accommodate the smarts built into a system to make it user friendly, the dimensionality of the design problem grows exponentially. As systems get smarter, they depend upon more information to operate. This implies interfaces with other systems in the Global Information Enterprise (GIE) - to request and obtain information, and to publish and subscribe to meet shrinking time requirements.

At the same time, breadboards and brassboards are becoming very expensive to build. Fortunately, they are becoming less important due to the software nature of these systems. Much of the guts of a system today lay in huge sets of imbedded algorithms in general purpose processors that provide the smarts. Testing these algorithms with all of the external interfaces they must support is becoming most difficult and expensive. To meet these challenges, a new approach to designing and testing these complex systems has evolved. This approach uses simulation.

2. INTRODUCTION

This is the Final Report from PSI on our participation in the first round of effort on the formation of the GIESim Laboratory. Since the beginning of this effort the goals of the GIESim effort have remained the same. GIESim must be capable of predicting the end-to-end performance and survivability of globally distributed information exchange and management applications, such as the Joint Battlespace Infosphere (JBI), Deployable Theater Information Grid (DTIG), and Information For Global Reach (IFGR). It is aimed at providing a powerful and dynamic generic modeling and simulation framework as a baseline for continuing simulations of future instantiations of JBI and other applications.

In PSI's view, such a simulation facility can be used to support development and test of many systems in various stages of their evolution, e.g., requirements analysis, system design, interface design, system testing, and support for design and test of system upgrades. In effect, GIESim can provide a laboratory for defining, designing, and testing complex components of the GIE. It can support live field testing by helping to plan tests as well as augment and extend test capabilities beyond what is achievable with the actual hardware. Given that simulations have been validated, they can support the interpolation and extrapolation of limited amounts of test data, a major factor in system evaluations and decisions.

3. BRIEF STATEMENT OF THE PROBLEM

The basic "problem" that GIESim is aimed at remains the same, and the efforts in the first round of work have served to define and expand the frontiers of Modeling and Simulation (M&S) associated with GIESim. A key element for achieving information superiority within the Global Information Enterprise is the development of a modeling and simulation environment to support analysis and synthesis of pertinent concepts. This environment must be dynamic, malleable, and sufficiently detailed to support clients who need answers to questions on the GIE as it evolves and becomes more complex. This environment will consist of both COTS and GOTS elements, and support a multiplicity of requirements.

A predictive framework needs to be established to ensure that battlespace information platforms are supported with required communications technologies. This framework, embodied in the GIESim lab, must be capable of predicting end-to-end performance and survivability of globally distributed information exchange and management applications, such as the Joint Battlespace Infosphere (JBI).

What will it take to ensure the success of GIESim? What does success imply? Based upon prior experiences in this area, PSI offers the following thoughts. One can envision a simulation laboratory where program managers sign up to make use of the facilities. They are motivated because they can save precious time and money getting answers to complex technical questions. They can use the facility to demonstrate the level of operational capability of systems under test. The results obtained can be validated by targeted testing and in-depth analysis. Most importantly, this laboratory evolves to be a reliable proving ground to support decisions on fielding systems. It also provides a repository for knowledge of what it takes to ensure successful use of R&D funding.

Given that the above vision is desired, what will it take to ensure its success? Success will be measured in the eyes of the beholders - the users and the decision makers for funding. If the JBI program is the important first user, what is needed to ensure its successful use? What is the JBI program looking for in terms of a simulation environment? What investments are needed to ensure JBI can make good use of GIESim? Can these investments be justified for JBI alone? If not, how can they be leveraged with other programs? What are the milestones, time frames and resources required to ensure the success of GIESim?

To answer these questions, PSI put together a suggested plan, and offered a set of steps as part of the plan to accomplish the objectives thus far set forth in this program. These steps were listed in prior monthly reports.

4. REVIEW OF WORK DONE

This Final Report summarizes work done since the September 18, 2002 GIESim meeting held at AFRL in Rome leading up to the final GIESim meeting held on December 4, 2002. This Final Report also addresses the work that culminated in a research proposal for 2003. The three major items that were requested for development for the December 4, 2002 meeting are as follows:

- **Comprehensive, final briefing for GIESim**. This final briefing from PSI reflects the inputs from other team members, and focuses on several key aspects to ensure success of the GIESim effort. Our final briefing is included as part of this report as Attachment I. Topics covered in the final briefing are listed below. <u>Section 5 of this Final Report provides details on the briefing</u>.

    - Potential Users and Applications requirements.
    - Generic Infrastructure Requirements.
    - Generic Approach to Model Architectures.
    - Generic Approach to Simulation to cut time and cost of realizations.
    - Suggested Work Items for '03 and '04.

- **GSS Attributes Applicable to GIESim**. This summary of GSS attributes was requested and was intended to describe the features of GSS relevant to GIESim. This summary presentation is included in this report as Attachment II. <u>Section 6 of this Final Report provides details on the attributes of GSS to GIESim</u>.

- **<u>GIESim Multi-Computer Demonstration Simulation</u>**. This demonstration was requested of PSI and consisted of three Laptop computers interconnected via HLA and TCP/IP Socket interfaces. The overview slides for the demonstration are included in this report as Attachment III. <u>Section 7 presents a detailed description of the multi-simulation demonstration, screen shots of the four interconnected simulations, and presents details on the models and architecture of each simulation.</u>

5. FINAL GIESIM BRIEFING FROM PSI

This section covers the Final Briefing from PSI given on the December 4, 2002 meeting of the GIESim Team. The slides from the briefing contain a great deal of detail. Consequently, this section will serve to overview the slides and will provide some additional narrative. Frequent references will be made to specific groups of slides.

PSI is committed to the success of the GIESim Lab and the GIESim Team. Our Final Briefing is aimed at ensuring this success. It is the culmination of the prior work done by PSI with respect to the GIESim effort and draws on our experience from other projects and from interactions with members of the GIESim Team and with leadership of AFRL/IFGC.

PSI chose to take a tool-independent or tool agnostic approach in preparing this briefing rather than focus on PSI specific tools. Some slides use screen shots from GSS simulations simply for the purpose of illustrating different points. Our goal was to explore specific topics with an eye towards the success of the GIESim effort. Our briefing considers potential users and applications of GIESim Lab, then considers infrastructure requirement, then approaches to model architectures, and approaches to simulation realization that can cut time and cost. Finally the briefing concluded with some suggestions for FY03 and FY04 work.

A main goal of GIESim is to predict the end-to-end performance and survivability of globally distributed information enterprise applications. To accomplish this, GIESim plans to evolve a simulation environment that can build complex simulation of communications systems by selecting from the most appropriate models and simulations from disparate simulation environments and platforms. The Final PSI Briefing considers and reflects on these goals, and takes a generic look at how to ensure success. By generic we imply approaches that are independent of specific tools and implementations.

5.1 POTENTIAL USER/APPLICATION REQUIREMENTS

The first section of our Final Briefing considers potential users and applications of GIESim (Slides 6 – 15). Understanding the potential programs and users of GIESim can help define its requirements in terms of capacities, performance, connectivity, etc. By potential programs we mean large programs like the Global Strike Task Force and its associated components like DTIG and AT AOC (Slide 6). Each program will have its own set of needs for the types of simulated communications solutions possible with GIESim. By potential users we imply organizations like AC2ISR and ESC in the Air Force, and DARPA and USSOCOM in the DoD (Slide 7). Each will have unique needs for the GIESim.

An investigation of potential programs and users will allow us to understand and size the needs of all program stages, from requirement analysis, through system analysis, design, development, and test in terms of connectivity, capacity, assurance, etc. (Slide 8). This analysis can (and should) drive the architecture and approaches taken to realize the GIESim Lab and any tools developed for it. For instance, if we consider modeling DTIG (Slide 10) we see that its communications infrastructure supports the JBI, which in turn will support other systems although in some cases the communications infrastructure directly supports the ISR, C2, and weapons. By asking questions associated with different applications and types of needs we can build a stronger set of GIESim capabilities. For instance, in the AT AOC, JBI will fulfill particular AT AOC requirements, and in turn, JBI will require specific support from the communications infrastructure (Slide 11). Other applications and scenarios such as Time Critical/Time Sensitive Targets (Slide 12) impose tight control loop requirements on the GIESim, and ISR imposes requirements to support a wide array of sensor types and traffic (Slide 13) and sources (Slide 14) that must be modeled in sufficient detail. For an Integrated Air Defense System (IADS), a GIESim simulation must be capable of playing realistic scenarios and sufficiently detailed communications traffic to measure the effectiveness of the communications systems (Slide 15).

5.2 GENERIC INFRASTRUCTURE REQUIREMENTS

What sort of generic infrastructure is required to support the GIESim? These are the kinds of questions and respective answers addressed in this section of the briefing (Slides 16 – 22). The space of simulation size and simulation complexity is potentially huge. Systems of very small size and complexity can be "modeled" by Excel spreadsheets. Systems that involve complex functionality such as electromagnetic propagation require complex and sophisticated models. This is shown on the horizontal axis of Slide 17. The other dimension of this chart is size as measured both by the number of different types of entities and the shear numbers of entities, such as a large number of LANS. The types of simulation solutions that GIESim is aimed at involve both types of complexity, e.g., large dynamic sensor-to shooter networks involving complex operational scenarios.

With the above perspective, we can look at a generic process to support simulation needs (Slide 18). There are requirement for both people and infrastructure (Slide 18 - 21). Customers want answers to questions posed to analysts. Analysts could potentially runs simulations directly, although in most cases they would rely on high level modelers, who in turn draw from a models library and who may rely on detailed modelers to build needed model. Customers may not always know exactly what they need from a simulation and may require support from a simulation expert to help them pose their question(s). Various types of experts may be needed to help define things like:

- the overall communications infrastructure being modeled for a particular application including its IERS and performance needs (all this could potentially involve multiple experts),
- detailed systems requirements for the various communications elements, e.g., LAN characteristics and radio parameters,
- overall scenarios for the application, and
- operational details of the mission(s) being planned.

In the near-term, large simulations will still require detailed modelers and input from various experts. Over time this situation will evolve just as the world of PCs has evolved (the “Wintel” paradigm). As the infrastructure of the GIESim grows, ease of use of the GIESim should increase with it. The ultimate goal is a “walk-up” GIESim facility in which the GIESim guides a customer through a definition process and then the “smart” system builds the required multi-simulation system and required scenarios automatically. Before this can effectively happen, inputs from potential programs and users must be considered and evaluated to direct architectural requirements, and capabilities of different simulation platforms and associated models will need to be codified in such a way that they can be selected on the basis of capabilities, abilities to interface, e.g., by HLA, etc.

5.3 GENERIC APPROACH TO MODEL ARCHITECTURE

Customers will be attracted to GIESim if it provides simulation solutions that meet their needs <u>when they need it and at a lower cost relative to alternative approaches</u>. Slides 23 through 29 address generic model architecture requirements to support this position. Model architectures are critical to achieving this goal, and must be designed in such a way that they are independent of particular tools. The use of symbolic, hierarchical modeling that uses icons to represent models along the lines of the physical characteristics of the system components is essential. This approach stands back from particular implementations and addresses core functionality. The GIESim Team needs to look at all factors to ensure the validity of an assembled simulation, including model accuracy, validity of input scenarios and data, and measures of merit for the results (Slide 25).

There are a host of factors that need to be considered to ensure validity. These are presented in Slide 26 and must to be considered when building and assembling models for a particular application. These factor raise important questions about assessing existing models and simulation platforms that may be candidates for use in GIESim. Furthermore, the dynamics introduced by different complex scenarios may require tailoring of certain models and may dictate different aggregations of models architecturally across different simulation platforms to account for overall performance demands and for the characteristics of the HLA fabric used to interconnect the simulations. (See Slides 27 – 29)

5.4 GENERIC APPROACH TO SIMULATION TO CUT TIME/COST

The objective of the GIESim capability is to provide solutions to customer needs for complex simulation solutions that are easy to use, and that are available when they need it and for a cost that is low compared to alternatives. The customer has problems to solve and wants to save time and money! This implies several important factors that are addressed in Slides 31 – 40. To achieve these goals, GIESim needs a generic approach to tailoring models rapidly from a growing library of models, it needs to be able to build complex scenarios fast, and it needs to be able to add new models to the library fast. The first two steps should not require experts. The final step will require subject-area experts, and the system should provide tools to assist in the model development process, such as the use of interactive graphics to build required models. A large, and growing library of models and simulations will be important to satisfying customer needs, as will a framework for assembling models and building them into simulations. Models of large communications sub-systems such as information systems, wireless systems and backbone switching systems are required. Graphic visualization and interactions will be important. Moving platforms, terrain, foliage, and things like C2 mission threads are examples of the types of models that are important.

5.5 SUGGESTIONS FOR FY03/FY04

PSI gave consideration to potential work in FY03 and FY04 (Slides 42-45). Since GIESim is still in the concept stage, PSI feels it is important to bring in users to assess their needs. This will help direct the evolution of GIESim and lead us towards development of architectures, tools, and expertise most supportive to eventual client satisfaction. This early exposure to users will help advertise the work and goals of the GIESim Lab and serve to aid in the growth of our infrastructure. A goal for GIESim is to eventually automate much of the synthesis of the multi-simulation environment including selection of models, connections through HLA, and compilation of scenarios, etc. We can test out some our precepts for GIESim, such as what is required to create a truly "walk-in" simulation environment. Early user contact will also clarify their expectations, needs, biases, timeframe, affordability, etc., and may disclose unknown requirements and unexpected considerations.

Use of existing simulations to demonstrate ease of complex analysis with complex scenarios is important to give the GIESim effort credibility and substance. It will also help to grow the model and simulation base for the GIESim infrastructure. Exposure to users and subject matter experts will help to shape and speed the evolution of the GIESim Lab. There is also an opportunity to build prototype demonstrations that draw from heterogeneous simulation platforms. This can test some of the fundamental concepts inherent in the philosophy of GIESim, and should identify challenges for early solutions.

PSI feels it is important to get the word out to users on what GIESim is building, and to provide users with demonstrations of simulations as examples of what we are pulling together. These users include AC2ISR, ESC, DARPA, SPAWAR, etc. (See Slide 44)

Subsequent to the Final 2002 GIESim meeting PSI prepared a proposal that was oriented around the gathering of program and user needs. This proposal is included here as Attachment IV and is discussed further in Section 8 of this report.

Recently PSI was asked to take part in the 2003 development of a multi-simulation demonstration of a slice of DTIG. A proposal for PSI work on this effort was sent to AFRL/IFGC separately from this report.

6. GSS ATTRIBUTES APPLICABLE TO GIESIM

PSI was asked to provide an overview of GSS attributes applicable to GIESim at the final 2002 meeting. The slides prepared are included in this Final Report as Attachment II. In this section of the report we will expand on and discuss the attributes of GSS as they relate to GIESim.

Some background history on the evolution of GSS will provide some insights into the capabilities of GSS and its attributes. Prediction Systems, Inc. (PSI) was founded in 1974 as a privately held, owner-managed engineering technology company, specializing in modeling, simulation and Computer-Aided Design (CAD). The founders have deep experience in control theory, mathematics, software engineering and CAD. The company helps clients analyze, design, test, and evaluate the performance of real-time prediction and control systems. PSI's clients are typically developing sophisticated electronic systems. Models of these systems must contain sufficient detail to provide an accurate representation of their dynamics. With PSI's CAD environment, final software modules can be embedded in a large system model. As a result, the models become correspondingly complex. To meet client needs, PSI has developed an advanced set of tools that significantly cuts the time and cost to build real-time simulations and systems. PSI licenses these tools commercially.

Since 1982, PSI has concentrated on developing a new technology to support large scale discrete event simulation. This new technology is embodied in PSI's ***G**eneral **S**imulation **S**ystem (**GSS**)*. Measured by clients who currently use it, in terms of time and dollar savings, ***GSS*** is revolutionizing the approach to large system development projects. Because of continual investments by PSI, ***GSS*** currently enjoys a high level of quality and elegance as a product, one that is recognized internationally for its technological breakthroughs.

In the early 80's PSI was asked to build a detailed simulation for the Army of several hundred mobile EPLRS radios. EPLRS radios are highly complex at a level of approaching that of JTIDS radios. A large contracting company had made three attempts to build the simulation and had failed after spending large amounts of time and large sums of government money. Faced with this challenge, PSI determined that computing resources available at the time might require the use of a parallel computer to achieve the execution speed required by the customer. This idea and the approaches that stemmed from it set the stage for the evolution of GSS as it is today.

First, speed of execution was (and still is) a top priority of PSI. Second, to make optimal use of a parallel computer it is important to understand process interdependence. Processes that share data are dependent on that data and are therefore not independent. PSI realized that the ability to visualize process dependency was important, and that this involved the ability to visualize which processes shared which data resources. This led to the concept of separating data from instructions, and use of a CAD approach to graphically represent processes and data resources separately and to show process-data connections. Hence, speed and support for parallel processing are key attributes of GSS. Third, a simulation environment needed to support large numbers of complex entities. This means affective use of memory, and the ability to scale. These features are built into GSS.

To complete the EPLRS story, PSI built the EPLRS GSS simulation for the Army in far less time and for far less money than had been spent on other approaches. More importantly, the GSS EPLRS simulation worked and supported a simulation of several hundred radios, and it ran fast. In addition to the advances introduced with GSS, the EPLRS simulation used the Fast Propagation Prediction System (FPPS) developed by PSI that was based on the TIREM III model.

In the early 90s, the software base that constituted GSS was out of control. Relatively simple feature changes and bug fixes were taking an unreasonable amount of time, and required a staff of a dozen or more to handle. In this time frame, PSI had created another "product" in parallel with GSS called the Visual Software Environment (VSE). VSE uses the same CAD-like approach to design as used by GSS, and the same language except that it does not support a schedule statement required for simulations. VSE is intended to produce software programs whereas GSS is intended to produce simulations. VSE was developed in the same low-level language as GSS. PSI took a big leap, and decided to apply its own CAD software approach to the building and support of both VSE and GSS. In a boot-strap process, they used VSE to design its own replacement and eventually built a whole new GSS based on VSE. Today, 2-3 engineers support and extend GSS and VSE, and do so much more rapidly.

Throughout the 80's and 90's PSI continued to build and deliver complex simulations and planning tools to satisfied customers. In addition to GSS and VSE, PSI had developed an interactive run-time graphics system called RTG. This allowed user to interactively modify a simulation while it ran.

In the late 90s, PSI built an interactive Visual Development Environment (VDE) for GSS and VSE. Not only did VDE allow for the visual design of a CAD-like simulation architecture, it also allowed for a direct connection between the simulation and model architectures and the underlying rules and data resources. VDE also supported visual simulation and model design along the lines of the physical subsystems of a system, and creation of hierarchies of models.

Also in the late 90's, PSI ported GSS, VSE and RTG onto the Windows platform. The intrinsic platform independence of GSS, VSE and RTG made this a very simple effort.

The sections that follow mirror the slides in Attachment II on attributes of GSS and provide additional details.

6.1 EASE OF USE

GSS has a user friendly CAD interface that provides for ease of development and reuse of models. Our CAD approach supports design along physical lines, and the resulting simulation and model drawings facilitate understanding of the simulation. This is particularly important in large simulation since it helps to ensure validity. Our CAD approach is easily understood and models and simulations developed by one person can easily be understood and maintained by another person.

The run-time graphics capabilities of RTG support visualization of dynamic model performance, and user interaction with the running simulation. Dynamic visualization can be highly important during model and simulation testing since unwanted or unnatural behavior can be more easily spotted.

Also the ability to interactively modify a running simulation is an extremely powerful capability. It allows dynamic and rapid changes to a simulation that would require recompilation for other simulation tools. RTG allows a user to interactively try excursions to a scenario to understand the impact. In fact, with GSS and RTG the interactive capabilities allow a user to interactively build and modify entire networks using hierarchies of icons.

In fact the inherit ability of GSS/RTG to support hierarchies of icons, allows layers of details to be covered and uncovered as needed. This can dramatically simplify complex networks and supports uncovering and zooming into details whenever needed.

6.2 DEVELOPMENT TIME / RUN TIME

The CAD approach to simulation architecture and model design allow design along physical lines as mentioned earlier, and support rapid development of large, complex simulations. GSS uses a high-level language to describe process rules, data resources, and simulation control specifications. These are easy to learn and use, are focused on the creation of simulations and avoid the complexities and vagaries of lower level languages such as C/C++. The language used in GSS is oriented at solving problems, rather than software design nuances. GSS gets an engineer to a sound simulation solution faster than with other languages. Its basic CAD and architectural approach supports reuse which also speeds development.

As mentioned in the history of GSS, an initial and prevailing goal for GSS is optimal execution speed! GSS was developed by PSI for use by PSI to deliver simulation solutions to clients. Most of our clients demand speed, since a complex simulation is likely to be run many times during experiments to test different concepts, scenarios, etc. Our clients tell us that GSS runs 10 to 100 faster than our competitors. One NAIC application originally developed by a competitor took almost 30 minutes to load 350 nodes. The GSS version which replaced it loads 2400 nodes in 11 seconds. Also, with other simulation environments, model complexity can dramatically increase run times. Since GSS is optimized for execution speed (and scalability –

see further section), the use of detailed models is much less of a concern with respect to execution speed.

The founders of PSI have their roots in electrical design of complex digital circuits, and they were often involved with finding worst-case, optimal design solutions for highly non-linear and often tightly constrained systems. They developed, and have refined over many years, optimization facilities that have been incorporated into GSS. The Optimization Facility of GSS is almost entirely automated and is very easy to use. Simulations that involve many instances of complex models that are dynamically interacting in a complex scenario are inherently non-linear. Finding solutions to things like optimal flight paths for collecting SIGINT or for minimizing mission threats are quite common. The Optimization Facility of GSS makes these types of problems quite tractable. The designer can focus on the optimization problem definition based on simple rules and parameters built into GSS and GSS takes care of the rest automatically. This completely avoids a designer getting into the complex, potentially arcane, and very time consuming task of building an optimization facility. PSI has done it for them in GSS.

6.3 SUPPORT FOR MULTI-COMPUTER SIMULATION AND PARALLEL PROCESSING

As mentioned earlier, considerations and support for parallel processing were built into GSS from the beginning. The separation of processes, e.g., rules, from resources, e.g., state data, allow graphic inspect of model and process dependencies. This separation also allows the GSS system to "catalog" these dependencies into an internal database. This directly supports the allocation of processes to processors in a multi-processor environment.

In the late 1990s PSI built a multi-computer version of GSS and VSE under contract with the University of New Mexico (UNM) that was used in the Maui High Performance Computing Center. This version of GSS/VSE allows multiple simulations to run concurrently as one large simulation on multiple processors. This required that PSI extend its multi-processor resource coherence and simulation clock synchronization subsystems to support a tightly coupled network of processors. Also, PSI developed a cross-scheduling subsystem to simplify the modeling process and paved the way to support an SMP environment.

Recently, PSI won a Phase I SBIR contract with DAPRA on High Efficiency, Scalable, Parallel Processing Alternatives. The object of our research will be to tie the architectural database into a modified run-time environment that optimizes the allocation of parallel processors to processes.

GSS and VSE also have built-in support DIS and HLA interfaces to support operations with heterogeneous simulations in a multi-simulation environment. In addition, GSS and VSE support the simple use of TCP/IP sockets for exchanging data and control information. Furthermore, GSS supports the use of resources that are shared between GSS simulations on the same processor (intra-processor resources) and between GSS simulations that run on separate processors (inter-processor resources).

6.4 SCALABILITY

GSS is inherently scalable. It was originally designed to handle large simulations of many dynamic entities. GSS has kept up with client demands for higher capacities.

GSS currently supports millions of icons. While this may sound high, consider the figure to the right. Icons for three "simple" network nodes are shown. This simple network can represent many hundreds or thousands of entities. With GSS/VSE and RTG an arbitrarily deep hierarchy of icons is supported. In this case, if the icon for Network 3 is uncovered, you will see the underlying detail for this node. You can zoom into to more detail, and uncover successively deeper details. Assuming that each top-level node has the same underlying complexity, then this "simple" network requires over 1300 icons. Furthermore, since RTG is based on scalable vector graphics, you can easily zoom in by the factor of 1000 to see the lowest levels of detail in the three node network.

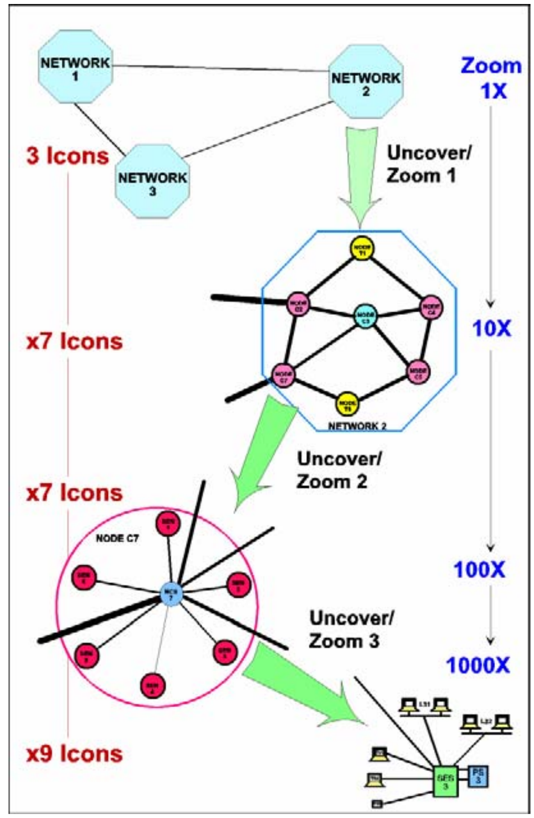


Figure 1 - Iconic Hierarchy

GSS uses Discrete Event Simulation and the scheduler has an event queue that currently supports 32,000 events. This can easily be expanded to over one million events. While this may seem like a huge number of events, consider an ad-hoc network consisting of several hundred mobile radios. Each radio can handle multiple computer-to-computer communications that can take place between multiple radios. This can easily create hundreds if not thousands of events.

GSS can easily handle thousands of complex entities. GSS has been used to model and simulate networks of hundreds of complex radios. In the TEL-SCOPE project for NAIC, GSS loads and builds a visual hierarchy of communications networks with thousands of links and nodes. The inherent capabilities of GSS make it very easy to migrate a simulation from a single processor to a multi-processor environment if needed.

6.5 DATA INTERFACE CAPABILITIES

GSS and RTG provide rich support for interfacing to data and to other systems. The figure below shows a current picture of the I/O capabilities of GSS/VSE/RTG. Items in green are under development and planned for the 11.0 Release of GSS, which is the next major release of GSS.

The Standard File Interface (SFI) was developed with several PSI clients, and provides a simple and powerful means to get data into and out of GSS simulations. An SFI file has a header that specifies the types and number of data fields that follow. Each SFI file is “connected” to one process. When the simulation is prepared, this header information in the SFI file is analyzed and compiled. When the simulation runs, the GSS system “automates” the process of reading the data contained in the file. The SFI approach greatly simplifies reading and writing large sequential data files. Multiple SFI files are frequently used to initialize model parameter data in simulations. SFI files are also frequently used to capture output data from a simulation.

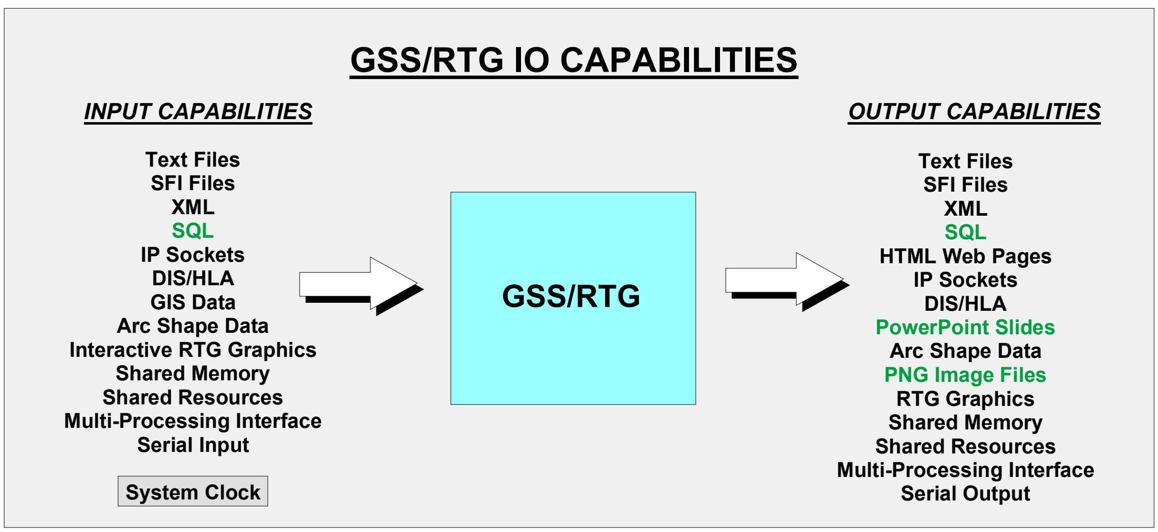


Figure 2 - GSS/RTG IO Capabilities

GSS also supports input from text and binary files. GSS file input can handle files with fixed and variable length records. GSS also provides facilities to open and close files for direct reading and writing of data. Files can be tested for existence and if they are empty.

GSS and VSE can read and write XML, and can generate HTML web pages. These features are currently in use on a project for the NAIC that is intended for near-term operational deployment. Development work is underway to support direct SQL interfaces to databases. This is a feature planned for Release 11.0 of GSS.

GSS is being used to input and output Arc Shape data and Arc Themes. This allows the “rendering” of GSS/RTG graphic output into a form useable by users of ArcExplorer. RTG output will also be captured to PowerPoint slides and in the form of Portable Network Graphics (PNG) image files. This work is currently underway and will be available quite soon.

As mentioned earlier, GSS and VSE support the use of TCP/IP sockets to connect to other programs and simulations to exchange data. The NFSS-OMS project that is described in a latter part of this report uses TCP/IP sockets to connect with external sensor systems. GSS and VSE also support built-in facilities to make the use of DIS and HLA interfaces quite easy. DIS and HLA have been used to interconnect GSS simulations and planning tools to external simulation environments on a number of projects.

GSS and VSE also support a number of built-in, shared resource types that facilitate data sharing between different GSS simulations or VSE programs. These essentially use shared memory in such a way that it is transparent to the GSS or VSE developer. GSS and VSE make sharing resource data between simulations extremely easy and natural. Different GSS simulations and/or VSE programs can run on the same platform or different platforms.

While not explicitly “data”, RTG supports dynamic, run-time interactive graphic user inputs, and supports dynamic output of results in graphic form. Inputs can be in the form of adding new iconic representation of models, in the form of lines that interconnect model icons, and movements and deletions of icons and lines. The system also supports the graphical user addition and connections of instruments, plots, etc. GSS and RTG also allow for the programmatic creation of icons, instruments, etc., and line types and styles are used to dynamically represent the states of things like communication connectivity.

GSS and VSE also support the use of serial I/O, and both can be synchronized to the system clock.

6.6 PLATFORM INDEPENDENCE

GSS, VSE and RTG are platform independent. They only require a ‘C’ compiler and OpenGL to run. OpenGL is a graphics standard developed by Silicon Graphics Inc. (SGI), and support for it is widely available. All of the buttons, lines, text, etc. used in the VDE of GSS, and in the GSS run-time environment are drawn using OpenGL. This means that the GSS platform and GSS created simulations look and act the same on all platforms.

All models, simulations, and icons built with GSS in one platform environment, e.g., Windows, can be exported and then imported to another environment, e.g., Linux. After being prepared on the new platform the simulation will behave and look exactly the same as on the original platform. The entire export/import process is extremely easy and quick.

GSS currently runs on Linux, Solaris, IRIX, AIX, and Windows NT4.0/2000/XP Professional.

6.7 CONFIGURATION CONTROL OF MODELS AND ARCHITECTURE

With GSS, configuration management is enforced by engineering drawings. These drawings can be annotated as needed with version and client information. GSS supports and encourages hierarchical modeling. Unlike many other systems, with GSS there is a direct relationship and connection between high-level architectural drawings and the underlying rules and data. The CAD-like drawing environment of GSS provides facilities to input revision history, current version number, ownership of revision, revision modification numbers associated with User Requests, etc.. An example of the type of information that can be provided is shown in the Figure below:

```
***PLOT LEGEND CONTROL INFORMATION
***LEGEND   :
***DATE     :
***TIME     :
***PAGE     :

***THE FOLLOWING ITEMS ONLY TAKE EFFECT WHEN LEGEND IS NOT SET TO NONE
***AND DEFAULT VALUES ARE SPACES

***COMPANY  :
***CONTRACT :
***MODEL    :
***DR       :
***ENG      :
***CHK      :
***APPD     :
***NEXT HIGHER ASSY :
***SIZE     :
***FSCM NO. :
***DWG NO.  :

*** MODEL USE AND DOCUMENTATION
```

Figure 3 - Simulation/Model Control Information

Each model subsystem supports the same levels of documentation in addition to the ability to add supporting text and user-level information for use by other model developers. A modeler can supply whatever additional information required/desired to describe the purpose and application of a particular model or simulation.

Also, the export capability of GSS allows different versions of models and simulations to be “archived” easily.

GSS also supports interactive, hierarchical use of icons. These can be used to define things like network configurations and platform equipment assignments at run time. Resulting changes can be captured to output files for later reuse.

6.8 COST OF OWNERSHIP

Measured by clients who currently use it, in terms of time and dollar savings, ***GSS*** is revolutionizing the approach to large system development projects. Because of continual investments by PSI, ***GSS*** currently enjoys a high level of quality and elegance as a product, one that is recognized internationally for its technological breakthroughs.

The cost of ownership of GSS is low for several reasons. First, GSS has very low licensing fees, particularly compared to other simulation platforms. GSS licenses are free for clients of PSI, and are provided on a sliding scale based on total contact dollars from a particular client.

GSS makes reuse of models and simulations easy. This leverages investments for development with GSS. Furthermore, GSS supports rapid development of models and simulations. Its architectural approach and rich graphics environment supports:

- **Model understandability** by a wide audience. From an economic standpoint, models that are more easily understood are more valuable because they are more easily validated, modified, and reused. The hierarchical design approach offered by GSS allows an analyst or other modeler to immediately understand the purpose of a model in the context of the overall simulation.
- **Model independence.** The ease in which one model can be replaced by another affects the economics of replacement. The design approach used in GSS allows the degree of model independence to be determined by visual inspection, and by designing along physical lines one can minimize interconnectivity between models.
- **High range of model validity.** The architectural design approach used by GSS, in addition to its execution speed, favors the design of models with more detail and sophistication. More detailed models tend to have a much wider ranges of validity. With GSS the investments required to build and test more detailed models is lower than with other simulation environments.
- **Interactive Graphics**. The run-time graphics (RTG) capabilities of GSS support visualization of model performance, and the ability to dynamically interact with a running model/simulation. This can speed development and testing of models by visualizing their behaviors. This can lead to rapid identification of bugs that might otherwise go undetected for a long time, perhaps even into production runs which are much more likely to occur with other simulation environments.

GSS is optimized for speed; GSS simulations run fast. This means reduced waiting time. Furthermore GSS may actually enable the use of simulations for conducting multiple experiments in a time frame that may prohibitive with other simulation platforms.

The Optimization Facility built into GSS eliminates the need to research and develop the complex algorithms and heuristics needed to find feasible and optimal solutions to complex, non-linear and constrained problems. Also, because the Optimization Facility is so easy to use, it promotes the use of optimization that would otherwise not be tractable or cost-effective with other simulation platforms.

GSS is highly scalable. This means that you can develop and test your models using small sample scenarios, and can be confident that your scenarios can easily scale to large numbers of complex entities. Other simulation environments work well for small numbers of entities, and then bog down and may even collapse as the number of entities approaches practical numbers.

Since GSS is platform independent and provides export/import facilities to quickly and easily move a simulation from one platform to another, there are virtually no “porting” costs involved with moving GSS simulations to new environments. Furthermore, simulation development and testing can be done on more generally available platforms, e.g., Windows, and moved to the operational environment, e.g., Solaris.

GSS also supports interfacing to external ‘C’ routes. This leverages the investments made for this “legacy” code, and provides a means to tie into proprietary routines and specialized software that a client may not to want to, or may be unable to rewrite into GSS. The latter case may occur if a client only has compiled object models.

Finally, GSS can dramatically reduce maintenance time and cost. “Life cycle” engineers typically need to learn the system they are maintaining, e.g., fixing and extending. The GSS design approach eases the cost of ownership transition, because GSS designs are intrinsically easier to understand than designs done in other simulation environments. Also, PSI does not charge maintenance fees to its client.

6.9 TRACK RECORD / EXISTING INFRASTRUCTURE

PSI has satisfied hundreds of DoD client needs for complex simulations and planning tools using GSS. A partial list of clients is shown in the table below.

DEPARTMENT OF DEFENSE	PRIVATE INDUSTRY
Defense Information Systems Agency	AT&T Business Marketing Group - FL
- Joint Interoperability Engineering Organization	AT&T Business Marketing Group - NJ
U.S. Air Force - AFCA, Scott Air Force Base	AT&T Consumer Product Division
- AFRL, Rome Research Center	AT&T Home Place Division
- AFIWC, Kelly Air force Base	AT&T Technologies
- NAIC, WPAFB	Atlantic Research Corp.
U.S. Army - Air Defense Center	BDM Engineering Services Company
U.S. Army CECOM	Booze-Allen Hamilton
- CAC2 Systems	Cincinnati Electronics
- Space & Terrestrial Communications	COLSA, Inc.
- EW/RSTA Center	C3I Systems Group, Inc
- I2WD	Command Control, Inc.
U.S. Army Research Laboratories	Dataproducts New England, Inc.
- Electronics Technology & Device Div.	EER Systems
- Survivability & Lethality Assessment Div.	Chrysler Corp. - Electrospace Systems
- Telecommunications Div.	GEC-Marconi Electronics Systems Corporation
U.S. Army - PEO Communications	(formerly Singer-Plessey)
- PM TRCS	GTE
U.S. Navy - Naval Research Laboratory	Hughes Electronics Company
	(EPLRS California Field Office)
FEDERAL GOVERNMENTS	ITT Aerospace and Communications Division
ATEA (Australia)	Jet Propulsion Laboratory
Canadian Marconi Company (Canada)	LOGICON
Data Sciences Limited (U.K.)	Lucent Technologies
(formerly Software Sciences Limited)	Maui High Performance Computing Center
FGAN - FHP (Germany)	MITRE
ISDEFE - (Spain)	PDC Company
TNO Physics and Electronics Laboratory	Raytheon
(Netherlands)	Sierra Cybernetics
NATO Communications and Information	Simulation Systems and Services Technologies
Systems Agency (Belgium)	(formerly Singer Link-Miles Sim. Corp.)
DRA (Royal Signals & Radar Est. - U.K.)	SRI International
	Stanford Telecommunications
EDUCATIONAL INSTITUTIONS	Teledyne-Brown Engineering
Iowa State University	TELOS
Monmouth University	TRW
New Jersey Institute of Technology	UNISYS
Nova Southeastern University	
University of Delaware	
University of New Mexico	
University of Toledo	

Table 1- Partial Client List

PSI has concentrated its client support efforts on tasks that capitalize on specialized company assets. These assets include comprehensive knowledge and state of the art tools for modeling and simulation applied to communication and control systems. These assets have been applied to all aspects of the analysis, design, test and evaluation process. PSI has placed the emphasis on solving long-standing problems in design optimization, estimation, line-of-sight determination and propagation path loss prediction using special algorithms for high-speed, high accuracy computation.

PSI has also been developing planning tools for clients for many years. At the heart of these planning tools are embedded simulations that test and verify the results of applying the planning tools – usually to complex systems involved in complex scenarios. Three recent examples include: TEL-SCOPE for the NAIC which is a network analysis tool for the information warrior, NFSS-OMS which is an operations management tool for complex sensor systems, and JTIDS Network Management Planning Tool which will enable much faster planning of JTIDS networks for complex scenarios such as North East Asia and which should result in an order of magnitude improvement in effective bandwidth.

PSI created GSS and RTG for their own use, and have extended and refined both over the years. Some refinements have been driven by client needs; others have been identified internally within PSI. The bottom line is that the infrastructure of GSS and RTG has been proven over and over again and refined as required across a very broad range of client applications and needs.

Recent contracts with DARPA on parallel processing and with AFRL Rome on PBA are serving to enrich and extend the capabilities of GSS, and the theoretical under pinnings of PSI's tool kit. Work with the Army I2WD is serving to enrich PSI's sensor and hierarchical sensor fusion capabilities. PSI plows client work back into its model and tool base on an on-going basis.

As a result, PSI has developed a large collection of models and simulations to draw upon. Some of these models are described in the next section including the multi-simulation demonstration built by PSI for the final 2002 GIESim meeting. That section shows drawings of the simulation architectures for four GSS simulations used in the demonstration – these will help one to understand the modeling and simulation environment, and some of the models in the PSI repertory. A partial list of simulations and models is presented in Figure 4 on the next page.

This list is intended to be illustrative rather than exhaustive. It is intended to show the breadth, depth, and range of the applications to which GSS has been applied, and types of models that are currently available. Within PSI, models are frequently reused and refinements are made as needed. As a result, PSI has a large base of accurate and validated models. Due to the overall approach taken by PSI in designing models and simulations, the models that are available are intrinsically flexible and are usually parameterized such that many types of related model behaviors are easily attainable simply use introducing a different set of initialization parameters.

SIMULATIONS (Partial List)	MODEL TYPES (Partial List)
EPLRS Simulation Facility	EM Environment
Asynchronous Tranfer Mode (ATM)	Fast Line-of-Sight (LOS)
Packet Network	Fast Propagation Prediction System (FPPS)
Circuit Switch	Foliage Loss
Local Area Net (LAN) Ethernet	HF Interceptibility
Distributed Queue Dual Bus (DQDB)	**Radio Models**:
ATM/Multiple DQDB/Gateway	*EPLRS*
Optimization Siting	*MSE*
Mobile Telephone	*SYNCGARS*
Mobile and Fixed Telecomm	*JTIDS*
Mobile Radio Comm Network Management System	Antenna
Dynamic UHF Radio Data Network Connectivity	Interference
Dynamic VHF Radio Telephone	Network Controls
RF Data Network Connectivity	MobilePhone - Radio Access Unit
Sensor Data Comm	**Switch & Router Models**
Packet Radio Comm Network	*Routers*
Single Channel Radio Network	*Central Node Switches*
Mutlichannel Radio Comm Connectivity	*Extended Note Switchs*
UHF Multichannel Radio Link	*Access Network Switches*
Air Defense C2	**Traffic Models:**
Fire Support C2	*Host Data Traffic*
Integrated Air Defense System	*Telephone Subscriber*
JTIDS/Link-16 Network Management	*Aggregated Subscribers*
WECM Radio	Sensors
WECM Graphical Interface	Hierarchical Sensor Fusion
UGS Sensors	**Host Platform Models:**
Ground Based Emitters	*Ground*
NFSS - Operational Management System (OMS)	*Surface*
Information Operations Planning Tool (IOPT)	*Aircraft*
Defensive Information Planning Tool (DIOPT)	*Satellites*
TEL-SCOPE	Movement Model
JTIDS Network Planning Tool	**Prototcol Models:**
High Throughput Terminal/CDMA Modem for SAT_COMM	*MIL-STD-188-220A*
MIL-STD-188-184 Emulation	*MIL-STD-188-184*
	MIL-STD-1553B
	Robust Transmission Protocol (RTP)
	FTAM, TP4, CLNP, HDLC
	TCP, UDP, IP, FTP
	Segment & Reassembly Protocol Layer
	GOSIP
	Link-16
	Pedistal Mounted Stinger Missile
	Air & Ground-based Jammers

Figure 4- Partial Lists of GSS Simulations and Models

PSI is has a large and growing client reference list, which is available on request. These clients span the National Air Intelligence Center (NAIC), AFRL at WPAFB and Rome NY, Air Force Information Warfare Center (AFIWC), US Army AMSAA, US Army AMSEL Research Site, and US Army CECOM C2D and I2WD.

7. GIESIM MULTI-COMPUTER DEMONSTRATION

PSI was requested to build a demonstration for the final meeting of the first round of effort on GIESim. This demonstration was intended to show how the capabilities of GSS can be used to support development for the GIESim Lab. A fundamental tenet of the GIESim Lab effort is to realize complex simulations of communications needs for use by larger force-level simulations by selecting and bringing together the most appropriate disparate simulations available required to solve communications modeling needs.

In the short time frame available, PSI chose to build a multi-computer simulation by combing three simulations built for the US Army I2WD program with an EBO IADS simulation built for AFRL Rome Research Site. The three I2WD simulations center around the Netted Full Spectrum Sensor (NFSS) system being developed by the Army. PSI initially won a Phase I SBIR for NFSS to develop a prototype Operations Management System (OMS) designed to manage multiple, disparate sensor systems deployed by the Army. Recently, PSI won a Phase II SBIR for the NFSS-OMS. The three simulations that were built for NFSS-OMS will be described in a section that follows. The simulation built for AFRL Rome Research Site was for their Effects Based Operations (EBO) program, and was for an Integrated Air Defense System (IADS). In the sections that follow, this report will:

- First provide a high-level overview description of each program, i.e., NFSS-OMS and EBO IADS. This section will also describe the limited modifications to each simulation required to build the multi-computer demonstration.

- This will be followed by a description of the multi-computer configuration including inter-simulation networked communications.

- The operation of the multi-computer simulation will be described next. This will start with the initialization sequence for the multi-computer simulation, and proceed with steps to interactively add sensors to the sensor simulation, followed by observing the impact on the IADS of detecting key C2 communications nodes. Selected screen shots of each simulation are also provided.

- Finally the models and architecture of each simulation will be discussed along with their respective drawings. This section is provided for the reader to develop a deeper insight into the GSS modeling and simulation environment, and to better understand some of the approaches PSI takes to architecture design and application of model re-use.

7.1 OVERVIEW OF SIMULATIONS

7.1.1 NFSS-OMS Simulation Overview

The NFSS-OMS is an on-going Phase II SBIR. The architectural design of the Netted Full Spectrum Sensor (NFSS) Operations Management System (OMS) is intended to manage and control disparate MASINT sensor systems via an integrated graphical interface. The NFSS-OMS will support connections with sensor systems for sensor report collection, processing, data fusion and graphical display. Interfaces to sensor control tasking will be provided using graphical panels. Sensor coverage maps will be generated automatically based on user specified parameters depicting the areas of the battlefield coved by each sensor system. NFSS-OMS uses PSI's state-of-the-art simulation and development tool, GSS and its non-linear Optimization facility, to provide mission planning for both pre and post deployment scenarios. Interfaces with other sensor management and control systems will be supported to expand sensor fusion and deployment deconfliction capabilities.

The NFSS-OMS uses two simulations as test drivers. A simulation of Unattended Ground Sensors (UGS) is used to simulate an actual sensor system that will be connected to the NFSS-OMS in operational use. The UGS simulation connects to the NFSS-OMS using TCP/IP sockets. In addition, PSI has developed a simulation of ground emitters that is used to feed the UGS simulation. The Emitter simulation interfaces to the UGS via HLA. The Emitter simulation also connects to the NFSS-OMS via HLA. This connection serves to test the NFSS-OMS. In actual operation there would be no connection between the NFSS-OMS and emitters. The NFSS-OMS also supports the interactive addition of airborne sensors "flying" interactively created flight paths. Key functionality of the NFSS-OMS includes:

Sensor Interface Server – The NFSS-OMS will provide a Sensor Interface Server for Sensor Systems (and simulations) to connect to. In addition to managing the sensor interfaces, it provides an important "normalization" service. This service takes the diverse data feeds from different sensor systems and puts them into a normalized format for sending to the Hierarchical Sensor Fusion Algorithm process.

Sensor Fusion Processing – This is a core function of the NFSS-OMS. A unique characteristic of the OMS approach is the use of hierarchical fusion. Here, the fusion process allows an operator to "uncover" and display layered sensor inputs that led to a particular fused sensor display or "spot report". This way, skilled operators can inspect the underlying sensor types and confidence levels leading to a particular fused result.

Interactive Visualization – A visual representation of sensors and fused data with a hierarchical approach are placed geographically on actual terrain data providing a valuable tool for understanding the battlespace situation and contributions to sensor fusion. Some visualization examples are provided in the section of the final report that describes the operation of the multi-simulation demonstration. The NFSS-OMS can display representations of emitters whose simulation information is provided for testing over HLA, and sensor data communicated over TCP/IP sockets.

Other Elements – The NFSS-OMS has many other subsystems that enable its overall functionality as a management system for I2WD. Some of these include:

- Sensor Resource Management – this module handles management of multiple sensor systems and ensures optimal use and deconfliction of sensor deployments.
- Sensor Tasking Driver – an interface to feed new tasking directives into the sensor systems. This might be to achieve better sensor coverage in some target region, or to augment one sensor type with another.
- Target Prioritization – this aids an operator in selecting and prioritizing targets.

For the purposes of the GIESim multi-computer simulation, the NFSS-OMS and the Emitter and UGS simulations use the same terrain data for Bosnia as used with the IADS simulation. GSS makes it very easy to input different sets of terrain data, and PSI has tools to automate the process of importing NIMA DTED terrain data. Other forms of terrain and background data, e.g., roads, can also be easily imported.

Also, for the purposes of the demonstration, a TCP/IP client was added to the NFSS-OMS to support a socket connection to the IADS simulation. This connection is used to report the fused sensor data to IADS.

7.1.2 EBO-IADS Simulation Overview

The Effects-based Operations Advanced Technology Demonstration (EBO ATD) required an Integrated Air Defense System (IADS) model with the following characteristics: immediate availability (within weeks, not months), end-to-end functionality though not requiring complete depth, malleable for R&D purposes, and able to integrate with existing and development models of other target systems.

The IADS model delivered by PSI met these requirements and included a scenario that demonstrated the desired capabilities. This EBO-IADS simulation is a proof-of-concept demonstration intended to show PSI capability in the EBO/IADS space, and is a means to present and dynamically interact with an integrated view of the battlespace. The PSI IADS demonstration can be extended in both breadth and depth. <u>The addition of a TCP/IP interface to the NFSS-OMS simulations to support the GIESim multi-simulation demonstration is an example of just such an extension of the EBO-IADS simulation.</u>

The scenario provided with the EBO-IADS simulation is reasonably complex and uses Bosnia as the theater of operation. The scenario contains movement paths (flight paths in this case); airplanes with UHF jammers, airplanes with early warning receivers and self-screening jammers; red force ground UHF networks, radar sensors, C2 units, and fire units with their own targeting radars; ground-based Coalition radios; Bosnia contour map; and a grid for referencing. Interaction of these elements with each other will be described later in this section. Dynamic interaction with the scenario within the GSS simulation environment is provided and will be discussed briefly later. “Elements” of the scenario provided are reviewed next, followed by a discussion on interactions between elements.

Elements of the Simulation Scenario

The simulation scenario included with the IADS simulation is defined by a number of input definition and initialization files. These files determine the number, disposition, and composition of each element. The actual interactions between elements are determined by individual models for each element and how they interact. Interactive modifications or additions to the running scenario are captured in output files that can later be used as new input files. Figure 5 shows an overview screen shot of the EBO-IADS simulation that shows many of the key elements in the simulation.

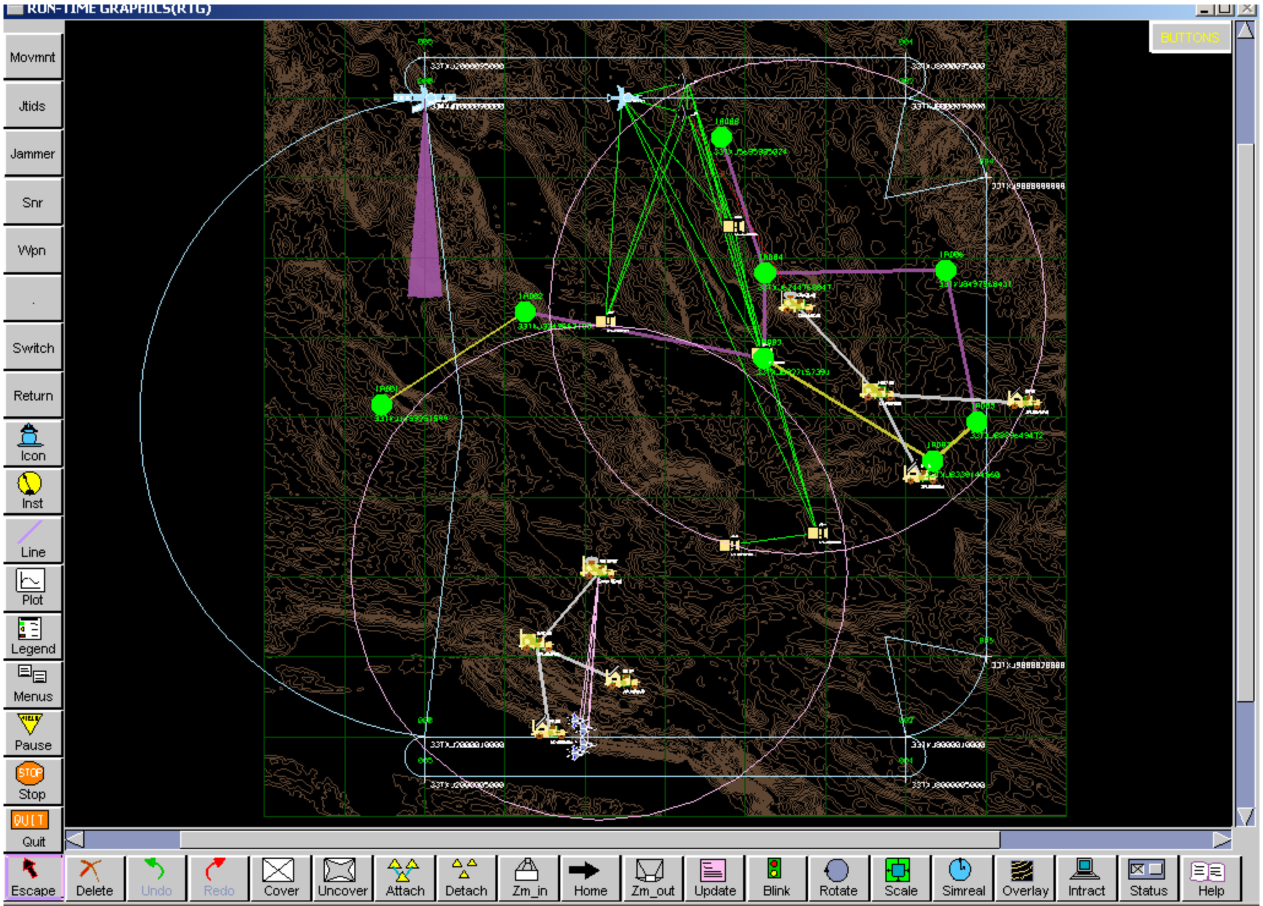


Figure 5 - EBO-IADS Overview Scenario Screen Shot

Bosnia Terrain

Bosnia was chosen for this simulation because of the complexity of it's terrain and frequent use in many wargaming activities and simulations. Rugged mountainous regions, broad plains, and bodies of water make Bosnia a challenge – particularly for radio and radar systems. While the simulation shows "flat" contours, the underlying data is fully 3D which is used in dynamic electromagnetic propagation calculations for the radars and radio. Recent advances in GSS/RTG support 3D perspective visualization of terrain. The IADS simulation can easily be changed to use some other terrain if so desired by modifying the initialization files, and adding the desired terrain map file.

Movement Paths

Movement paths and the ability to place equipment on paths and launch them is central to a simulation such as IADS and essential to any EBO embodiment. The IADS simulation has defined three paths: two narrow, elongated paths positioned on the northern and southern edges of Bosnia, and one, very large path that encompasses most of Bosnia. Coordinates of the waypoints that define paths can be optionally displayed. Paths can be modified interactively, and added as desired while the simulation runs. The PSI "movement model" is quite sophisticated and allows for a path to be defined by a large number of "waypoints" which have three dimensional coordinates. Equipment can be placed on paths with specific flight speeds along the path, and with specific "launch" times. Input files are used to initialize movement paths and to define assigned equipment. All IADS modifications are saved to output files for later use.

Coalition Aircraft and Equipment

The scenario provided includes a formation of four jets "flying" the southern path, a formation of two jets flying the northern path, and two large aircraft each equipped with a UHF jammer flying the large encompassing path. Jammers are represented by narrow triangles that are purple when enabled and white when disabled or off. Each aircraft is "equipped" with a radio (in this case a very detailed and accurate model of a JTIDS radio). The smaller jet aircraft can be equipped with optional early warning receivers and self-screening jammers.

Connectivity between JTIDS radios is shown as either green or curved dotted yellow lines. Either line is the result of a detailed propagation calculation taking the terrain data into account. Solid green lines indicate bi-directional communications connectivity, while curved, dotted yellow lines indicate one-way connectivity. The later situation can occur if one radio is in a high SNR environment that blocks it's receiver while it's transmitter can be heard by another radio in a good SNR environment.

Opposition Ground UHF Network

The solid green circles in Figure 5 represent UHF ground radios of opposition forces. Eight UHF radios are shown. Purple links between these radios indicate good connectivity, whereas red links represent UHF links that are down due to UHF jamming, and yellow (or gold) links represent cable links which are immune to jamming. The inclusion of these UHF ground radio is to demonstrate the effect of air-borne jamming in the model and simulation for IADS.

Opposition Ground Threats and C2 Network

Two opposition-forces sensor networks with associated C2 Units and fire units are included in the IADS scenario. One group is in the upper right quadrant of Figure 5, and the other is in the lower part of the terrain slightly to the left. Here, gray links indicate the UHF connection between the Radar Sensor, C2 Unit, and the Fire Units. The Radar Sensor presents the current air picture to the C2 unit for analysis, and the C2 Unit will pass Fire decision information to the Fire Units, who will in turn use this data to start "hunting" for targets. The Radar Sensors perform detailed radio propagation calculations, and the simulation takes into account the radar cross section (RCS) of aircraft, and can be impacted by self-screening jammers carried by aircraft.

Good radar tracks are indicated by solid purple lines and dotted yellow lines indicate the radar has just started tracking and not yet locked, and a dotted red line indicates that the radar has lost it's lock and is "coasting" the target by using previous data to compute it's most likely trajectory while it attempts to reestablish it's lock.

Coalition Ground Radios

For completeness the IADS Demonstration includes a number of ground-based radios. The radios are rectangular gold "boxes". Green lines between the two radios and the overhead aircraft represent connectivity between radios (in this case JTIDS radios).

Scenario Interactions

Interactions between elements in the IADS Demonstration scenario and simulation are quite "natural", dynamic, and automatic. Coalition electronic warfare forces fly jammer platforms against critical adversary telecommunications UHF networks in an attempt to reduce the ability of critical command and control systems to share sensor, weapon, and C2 information. Simultaneously, coalition air forces will fly offensive missions against adversary C2, weapon, and sensor ground targets.

Threat air defenses will employ active radars to search, detect and track both offensive and defensive coalition air forces. Selected platforms will employ early warning receivers and self-screening jammers as counter-measures against adversary tracking radars. Threat command and control (C2) systems will network to produce and exchange a composite air picture. Threat air defense weapon systems will evaluate synthetic air pictures to select, evaluate, and engage coalition air forces. Adversary air defense missile systems will launch active guided missiles at friendly aircraft attempting to destroy as many as possible so to minimize the damage inflicted by coalition electronic and kinetic munitions systems.

A user of the simulation can dynamically interact with the simulation as desired to observe resulting effects. Existing elements can be disabled and re-enabled, e.g., radar sensors, airborne jammers, and missile systems. Flight paths can be modified as needed. A user of the IADS simulation can also add elements dynamically by using the ICON function button of GSS/RTG. Aircraft can be added, equipped, attached to movement paths, and then launched. Ground-base elements can be relocated to evaluate the effect of terrain on a new position, e.g., better sensor placement on a higher ridge. Weapons systems can be re-oriented for more optimal coverage on an air "corridor".

7.2 MULTI-COMPUTER, MULTI-SIMULATION OVERVIEW

As discussed earlier, the demonstration that PSI built is intended to demonstrate the ability to interconnect multiple large simulations to test real world systems and tools in the GIESim Lab. The multi-simulation configuration is shown below.

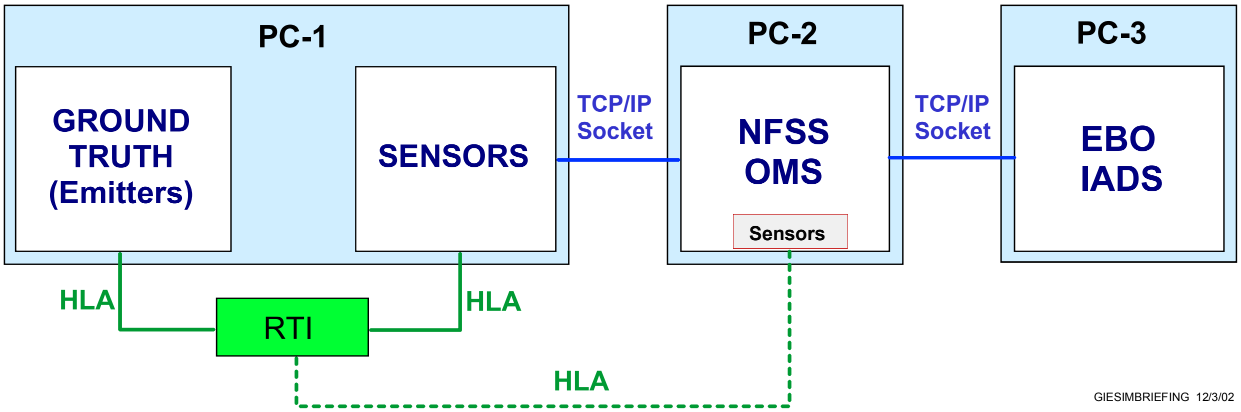


Figure 6 - Multi-Simulation Demonstration Configuration

The Emitter Simulation and Sensor Simulation associated with the NFSS-OMS were run from one laptop computer (PC-1). The NFSS-OMS simulation ran on another laptop computer (PC-2), and the EBO-IADS simulation ran on laptop PC-3.

The RTI ran on PC-1, and HLA connections were established between the Emitter Simulation, Sensor Simulation, and NFSS-OMS simulation. The HLA connection between the Emitters and NFSS-OMS was a test path, and allowed the OMS to display ground truth positions of emitters for comparisons with sensor estimates of emitter locations. In this case, the Emitters publish data that the Sensor and NFSS-OMS subscribe to.

TCP/IP sockets interconnect the Sensor, NFSS-OMS, and EBO-IADS simulations. The Sensor simulation is a client to the NFSS-OMS TCP/IP server process, and NFSS-OMS is a client to the EBO-IADS TCP/IP server process.

In this multi-simulation demonstration, the EBO-IADS simulation was modified to obtain the location of the Small Extension Node radios (SENs) that interconnect the C2 Units and the Radar Sensors. These Band 1 radios (350-400 MHz) run the communications network that links the Radar Sensor to the C2 Unit. Once the location of the SENs is determined, then they can be jammed by an airborne jammer to protect coalition jets in transit to a target. The Emitter simulation was modified to represent these specific radio emitters, and the NFSS-OMS was modified to feed fused sensor data to the EBO-IADS simulation for the purposes of the GIESim demonstration.

7.3 OPERATION OF THE MULTI-COMPUTER DEMONSTRATION

The multi-simulation demonstration is started in stages, starting with the initialization of the RTI. Next the IADS simulation is started. IADS initializes from scenario input files, and builds the interactive visual display. The associated IADS TCP/IP Server then starts listening for a connection to be established from the NFSS-OMS client. The initial IADS display is shown below.

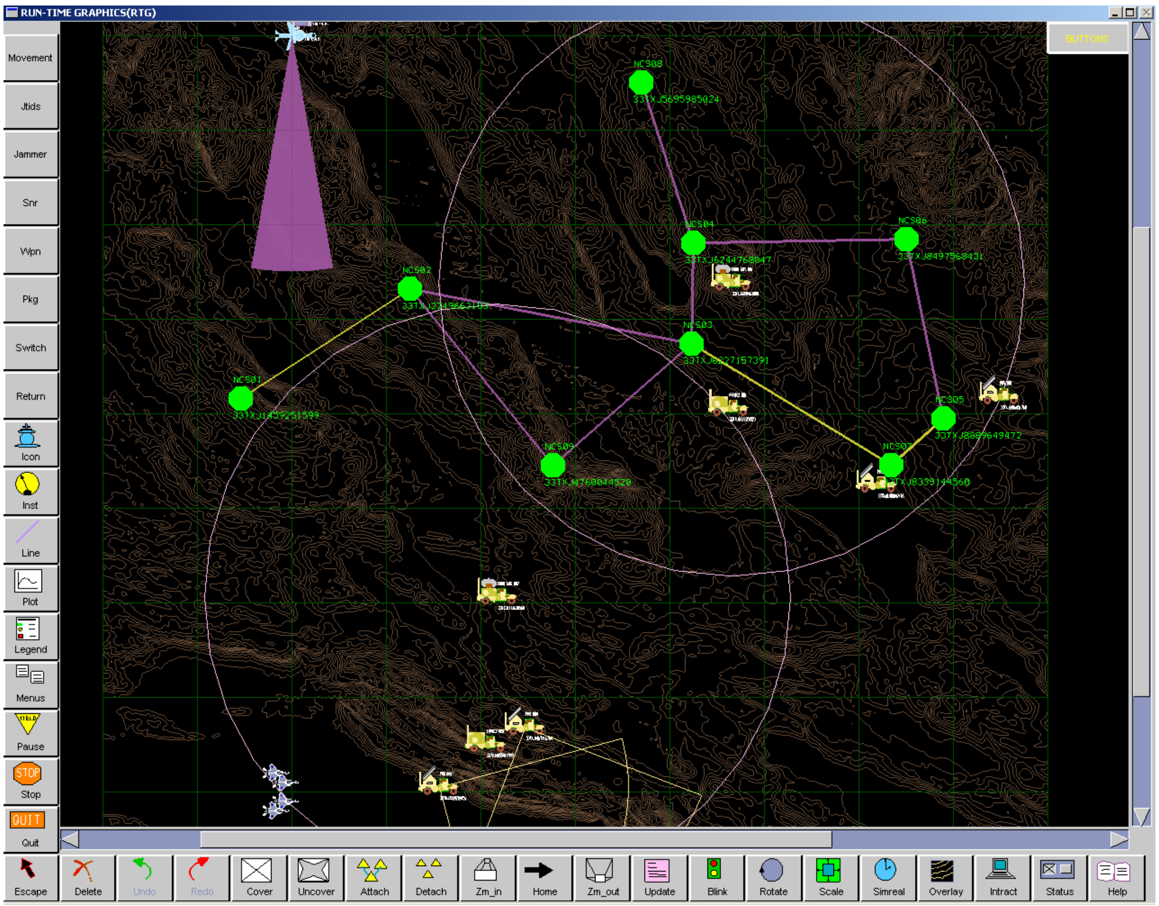


Figure 7 - Initial IADS Display

There is an escort jammer sitting on a runway at the top left of the display, and a squadron of four jets ready to launch in the lower left of the display. These jets need to fly through the Radar Sensor and Fire Units near the center of the lower edge of the display to accomplish their mission. IADS waits until the locations of the communication nodes that connect the Radar with the C2 Unit are reported by the NFSS-OMS. Once these location are reported the jammer will launch on a schedule that knocks out the Radar-C2 communications to allow the jets safe passage.

The NFSS-OMS simulations are started in the following sequence. The NFSS-OMS is started followed by the Emitter Simulation, and then the UGS Simulation. On the NFSS-OMS, small icons that look like radars are the exact positions of the UGS sensors as reported over TCP/IP and yellow dots are the reported positions of the Emitters. Each simulation environment displays the same terrain contours for Bosnia. Figure 8 below shows the display for the Emitter simulation. The lower two icons represent the radios associated with the Radar Sensor and C2 Units of interest in the IADS simulation. Blue antennas mean that they are currently transmitting.

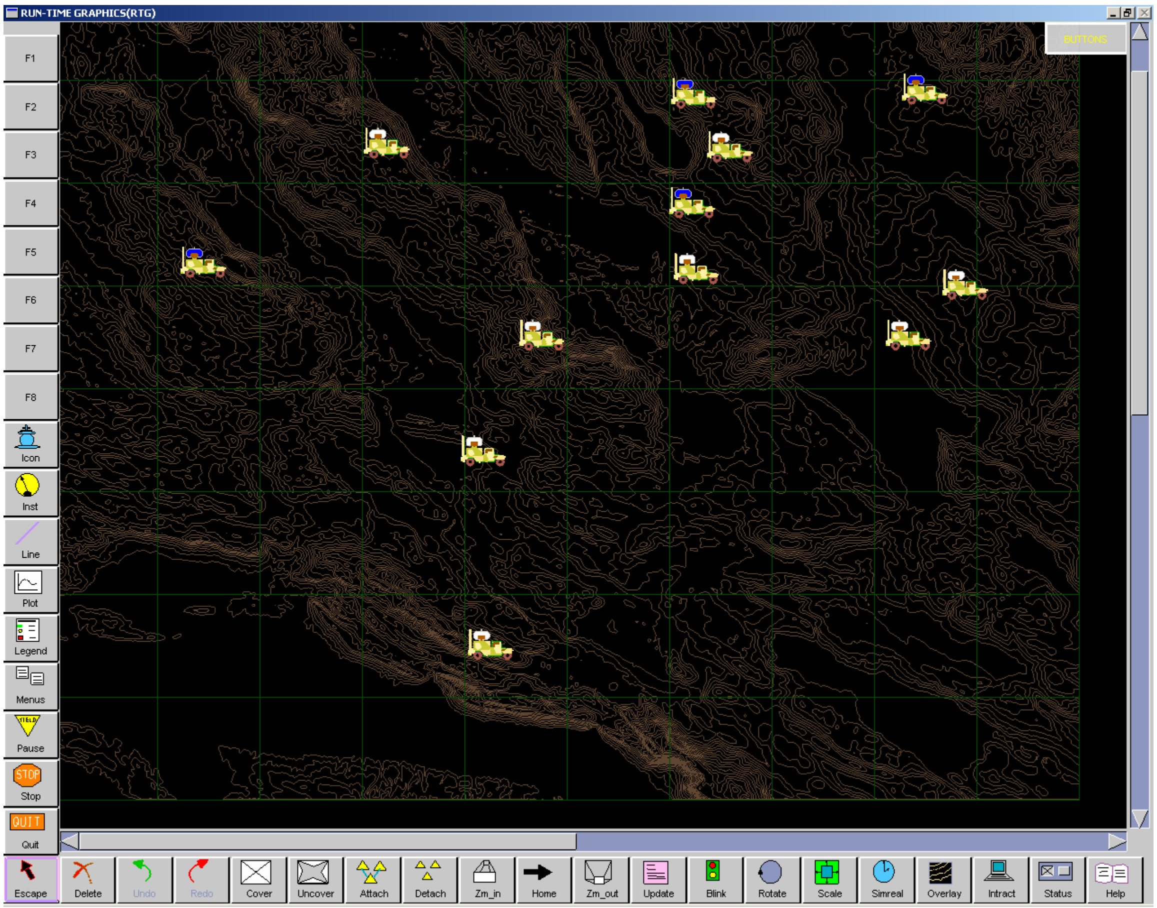


Figure 8 - Emitter simulation display

Initially the display of the UGS simulation just shows the terrain contours. UGS must be added to the simulation interactively by placing icons on selected places on the terrain. This is simply how the simulation was designed. Sensors could also be placed automatically during initialization from input files. UGS need to be placed such that the emitters of interest are detected and reported to the NFSS-OMS.

After placing a number of UGS on the terrain, the UGS simulation display appears as shown in Figure 9 below. The NFSS-OMS will now show the fused spot reports of emitters as collected by the UGS. The resulting NFSS-OMS display is shown in Figure 10.

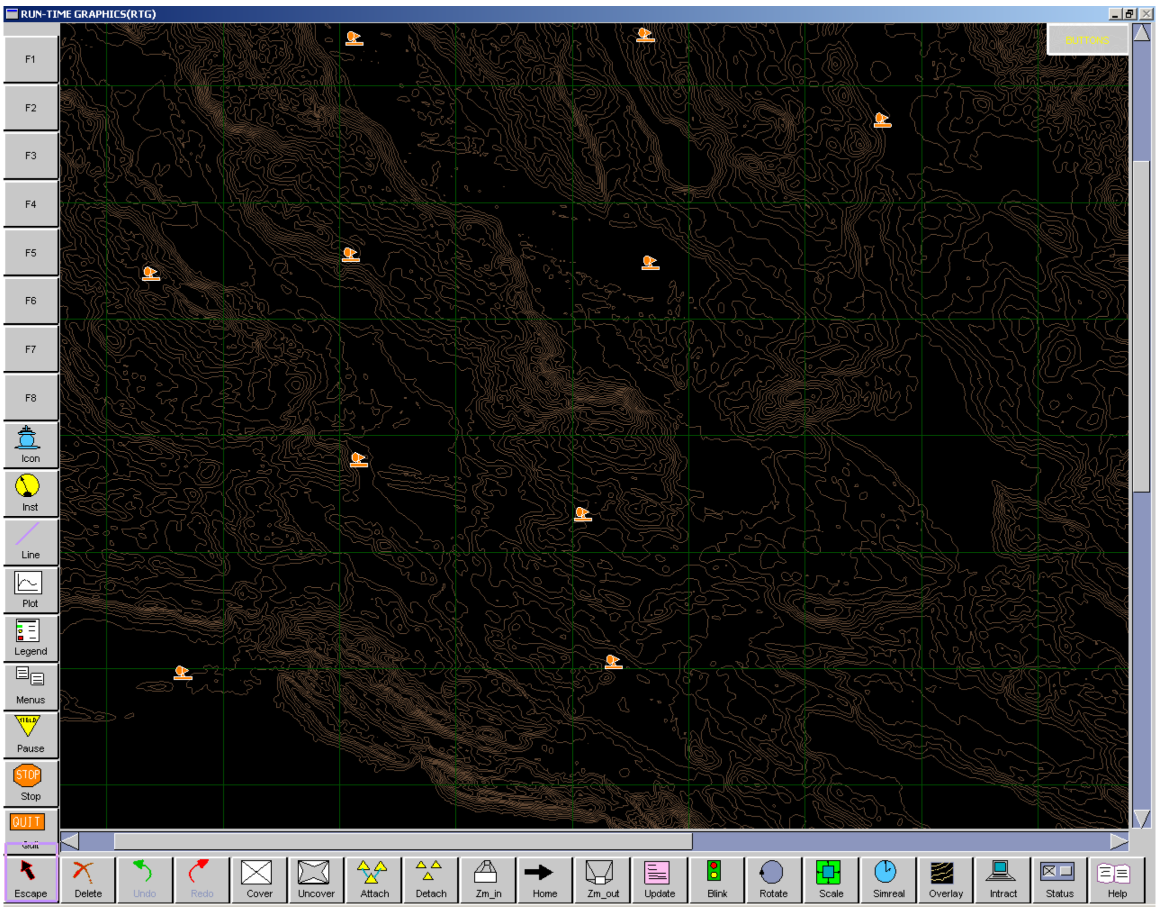


Figure 9 - UGS Simulation Display After UGS Placements

On the NFSS-OMS display, the UGS sensors are shown using the same icons as shown on the UGS simulation display above. Fused sensor spot reports of detected emitters are shown as orange disks. These spot reports are hierarchical in that each one can be uncovered to view the individual spot reports sent by sensors. The individual spot reports appear as small orange circles. As spot reports dynamically update, sometimes the individual spot reports appear momentarily. One example can be seen slightly above the center of Figure 10.

The NFSS-OMS sends its spot report data to the IADS simulation over the TCP/IP connection that was established. The IADS determines which emitters are the Small Extension Notes (SENs) connecting the Radar and C2 Unit, and then the jammers and mission squadron are launched on a schedule such that the jammer blocks communications from the Radar to the C2 Unit, which allows the tactical jet aircraft to pass through unharmed.

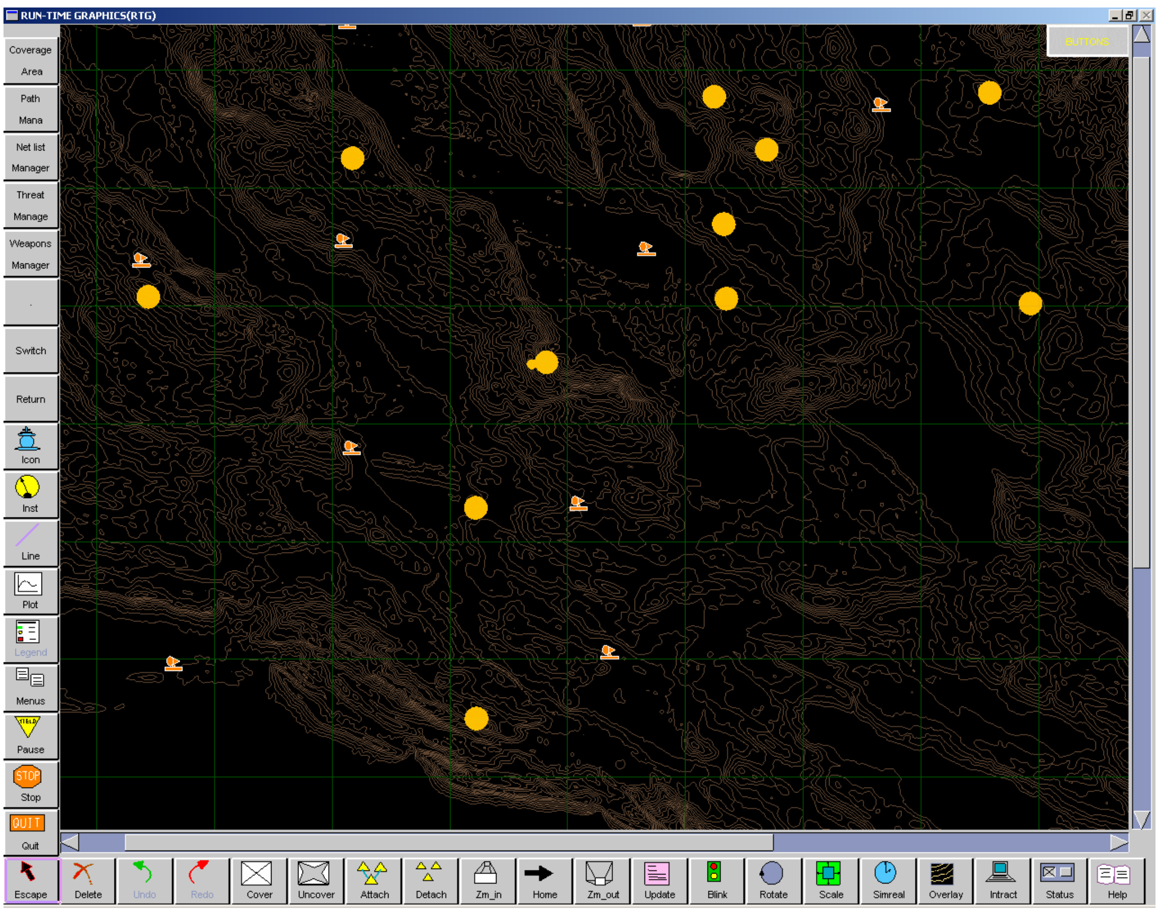


Figure 10 - NFSS-OMS with UGS Spot Reports and Emitters

The sequence of IADS screen shots that follow tell the rest of the story.

See the initial IADS screen shot below. The IADS Radar-C2 systems are linked by their respective SENs through the backbone network. The SENS (shown as yellow polygons) and their associated network connections appear after the NFSS-OMS reports their positions. As a result the IADS weapons systems become active as indicated by the displayed footprint of the targeting radars of the Fire Units. Note, the backbone network operates on a different frequency from the SENs radios and is unaffected by the airborne jammer.

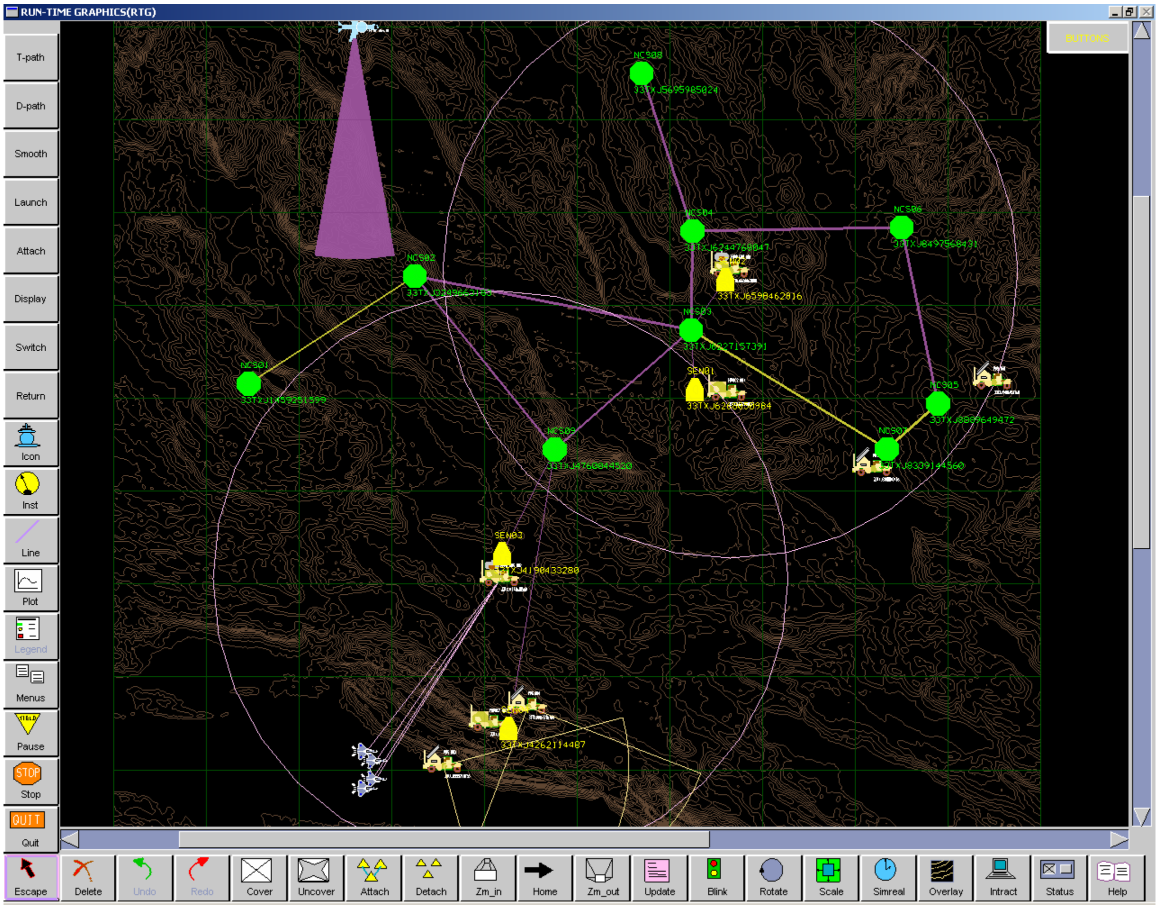


Figure 11 - IADS Just After Launch of Aircraft

Just after the launch of the aircraft, the attack aircraft are being tracked by the Radar Sensor as indicated by the solid purple lines. However, they are still out of range of the weapons system that they are heading towards. Also, at this point, the jammer is too far to the west to affect the key communications systems of this defense system.

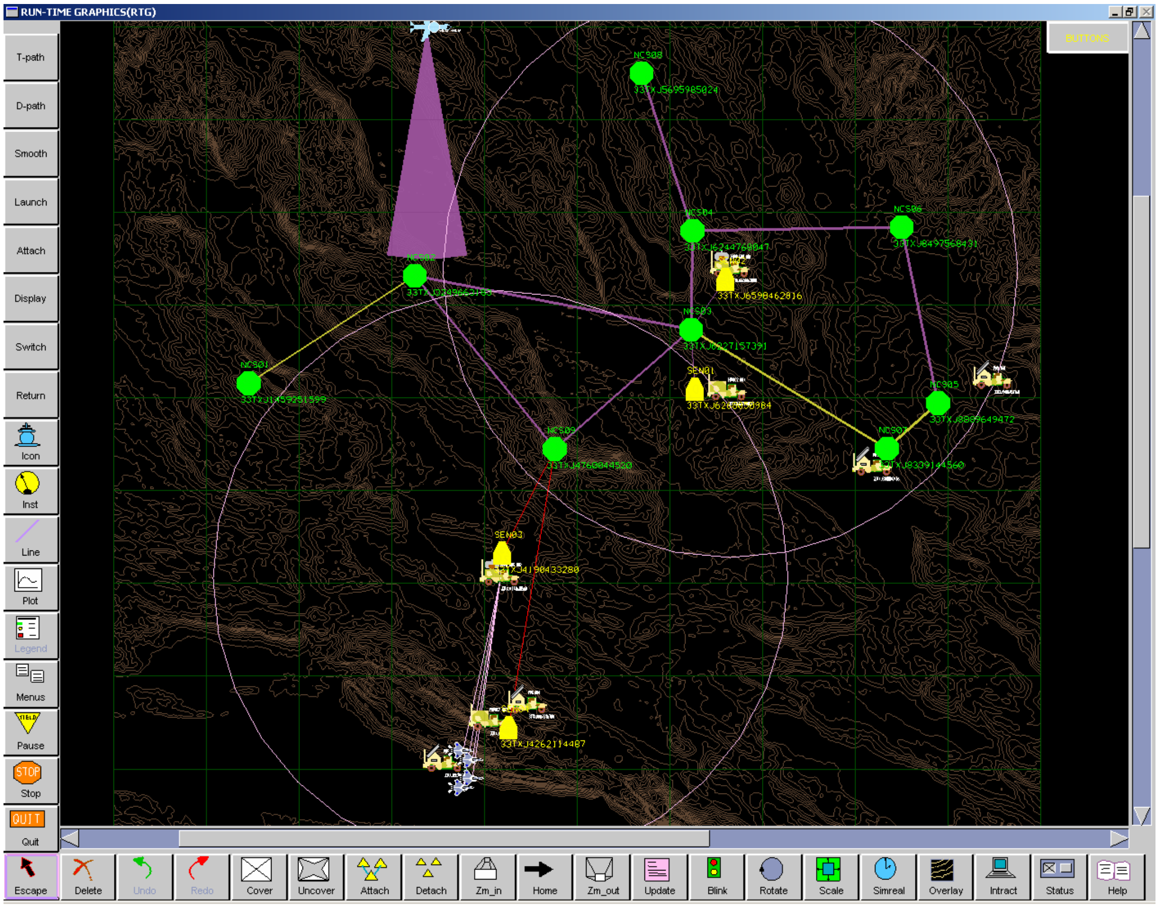


Figure 12 - Jammer knocks out Radar-C2 SENs Communications

As the jammer and mission squadron get closer to the weapons system, the jammer is in position to jam the Radar-C2 communications links – this is indicated by their links turning red. This results in the Fire Units standing down, which is indicated by the fact that the footprint of their targeting radars is no longer displayed. Note that the mission jets are still being tracked by the main IADS Radar Sensor.

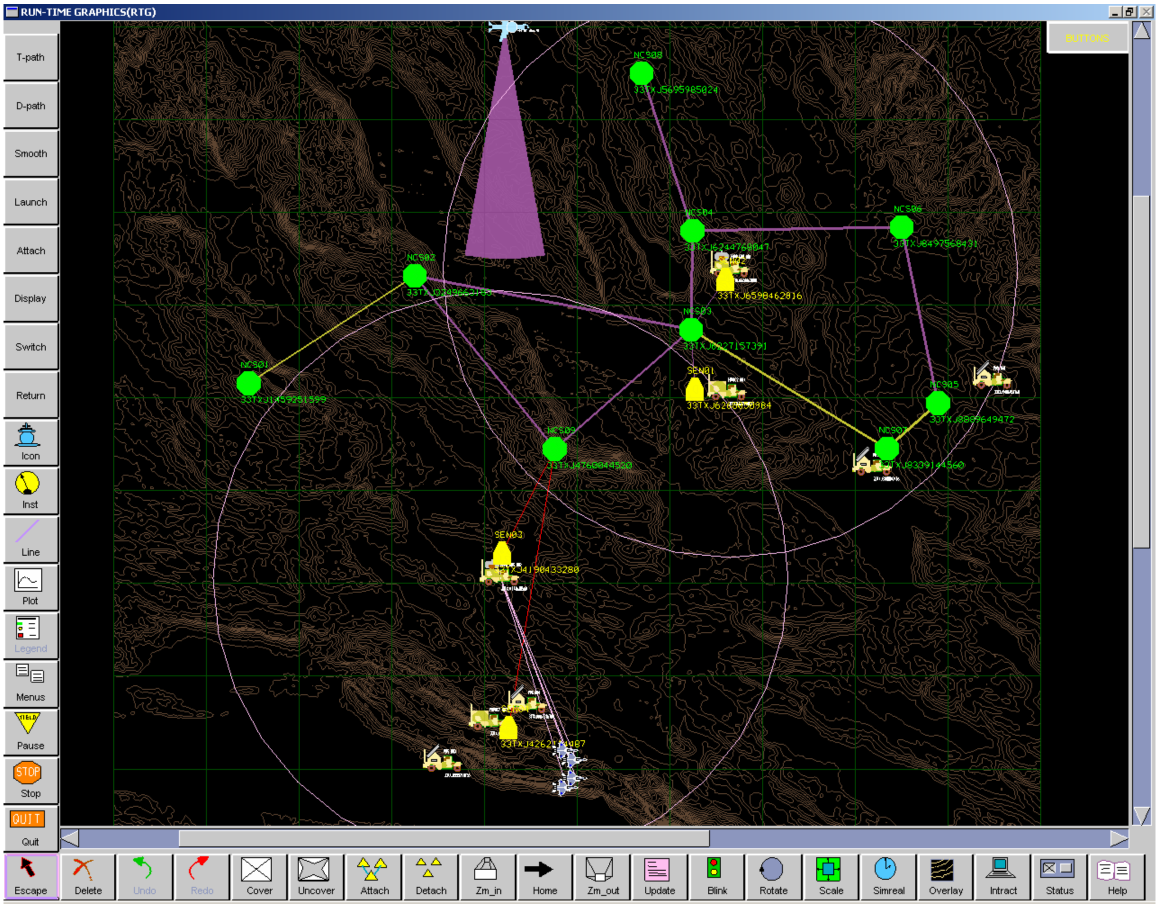


Figure 13 - Jets fly by IADS Weapons System

As time proceeds, the jammer and mission jets continue towards the east. The jammer is still blocking the SENs radios.

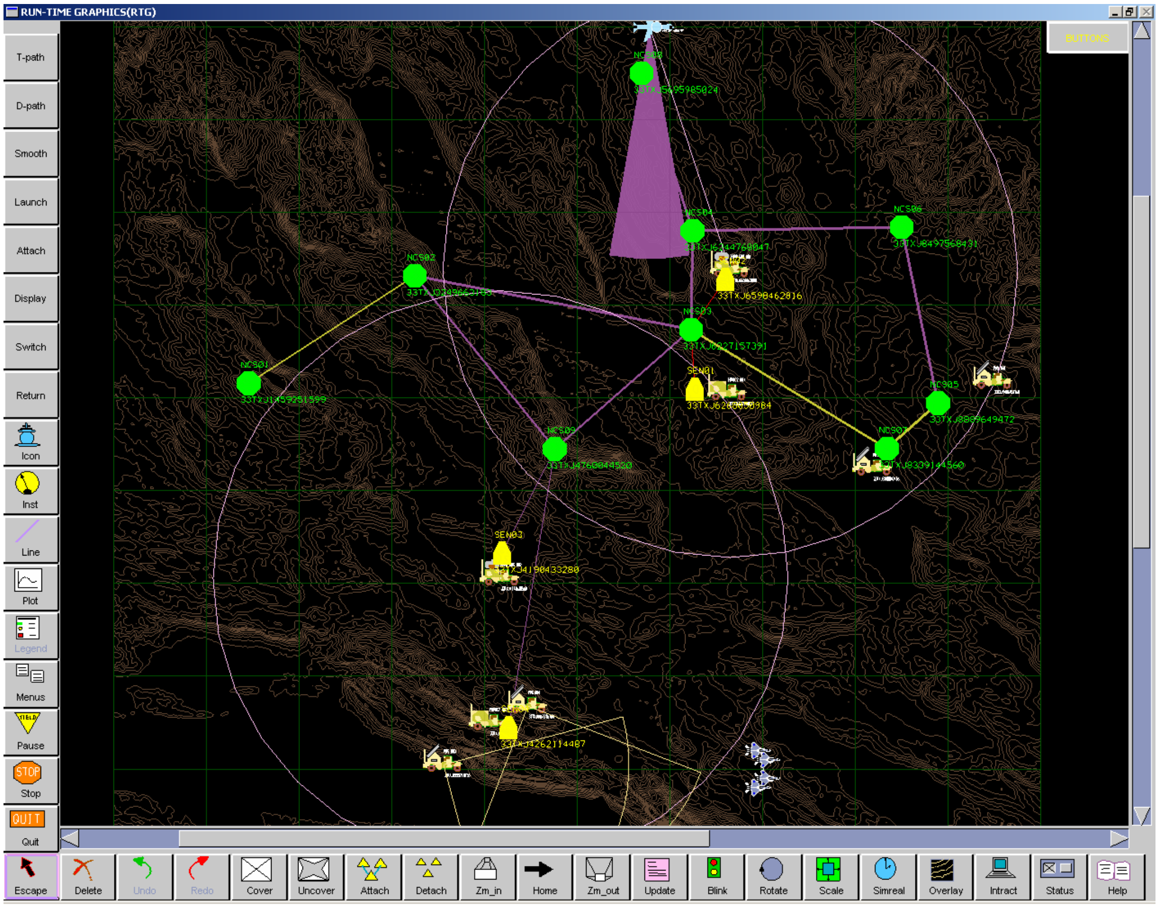


Figure 14 - IADS Weapons back up and Jets safe

Eventually the jammer moves far enough to the east such that it no longer affects the IADS weapons system. The targeting radars of the Fire Units are back on. However, the attack jets are well past and are out of harms way.

This whole sequence of simulations and their associated events was shown at AFRL Rome Research Site at the final 2002 meeting of the GIESim project. In addition, we also showed what happens to the coalition jets when they attempt to fly a similar course when the IADS is not jammed. They are very quickly shot down.

In summary, PSI was asked to build a multi-simulation demonstration to show the ability to interconnect multiple large simulations to test real world systems and tools in the GIESim lab. This section of the Final Report described the approach used by PSI, the architecture and configuration of the multi-simulation environment and showed representative screen shots from each simulation. The section that follows provides more detail on each simulation and presents a brief overview of their architecture and models.

7.4 MODELS AND ARCHITECTURE OF THE SIMULATIONS

This section will briefly review the models and simulation architectures of the four simulations used in the multi-simulation demonstration. As discussed in the section on GSS Attributes, GSS uses a CAD-like approach to modeling and simulation design. See Figure 15 for reference. At the lowest level, processes determine the behavior of models and use data contained in resources. Processes are shown as rectangles. State information is kept is resources which appear as rounded rectangles. Resources can only be accessed by processes to which they are connected.

GSS intentionally separates process rules from data, i.e., resources. This allows the detailed architecture to be drawn directly and allows visual intersection of interdependencies. Processes and resources are typically grouped together into "models". These models can be grouped together into higher level models forming a hierarchy. In GSS, model hierarchy can be as deep as needed. Models that only contain processes and resources are referred to as elementary models, whereas models that contain other models are referred to as hierarchical models.

Typically a designer starts with a high level architecture and proceeds to design along physical lines, and defines lower level models in successive stages of refinement until arriving at the lowest level where processes and resources and their interconnections are defined. At the lowest level, double clicking a process or resource opens an editor window to create and modify behavior and data respectively. <u>PSI's CAD approach establishes a direct connection between architecture and code and data.</u> When processes and resources that are widely separated need to be interconnected, the designer can connect then by using connectors (small circular, labeled icons); this is very similar to the approach used in electrical CAD drawings.

GSS uses icons to represent different input and output file types. GSS also provides support for utility models, which are used to provided frequently needed functions within a single simulation, and library modules, which are compiled and provide support for many different simulations.

Associated with each simulation drawing is a "control specification" that specifies things like names and locations of library files and certain data files. The control specification also specifies which models and processes start when the simulation begins, specifies an HLA interface handler, specifies graphics information needed including icon used, background overlays and a graphics event handler, specifies input and output data files, and specifies models used to evaluate the simulation at the end of a simulation run. The control specification is also used to specify how a simulation will be run: once, multiple times, or for optimization.

The sub-sections that follow will present the architectural drawing of each of the four simulations, and a brief description of the models and their function. The simulations are "uncovered" down to the level of elementary models. This provides a better view of the overall architectures. A fully uncovered view is presented for IADS.

7.4.1 Emitter Simulation

The emitter models and simulation architecture drawing is shown in Figure 15.

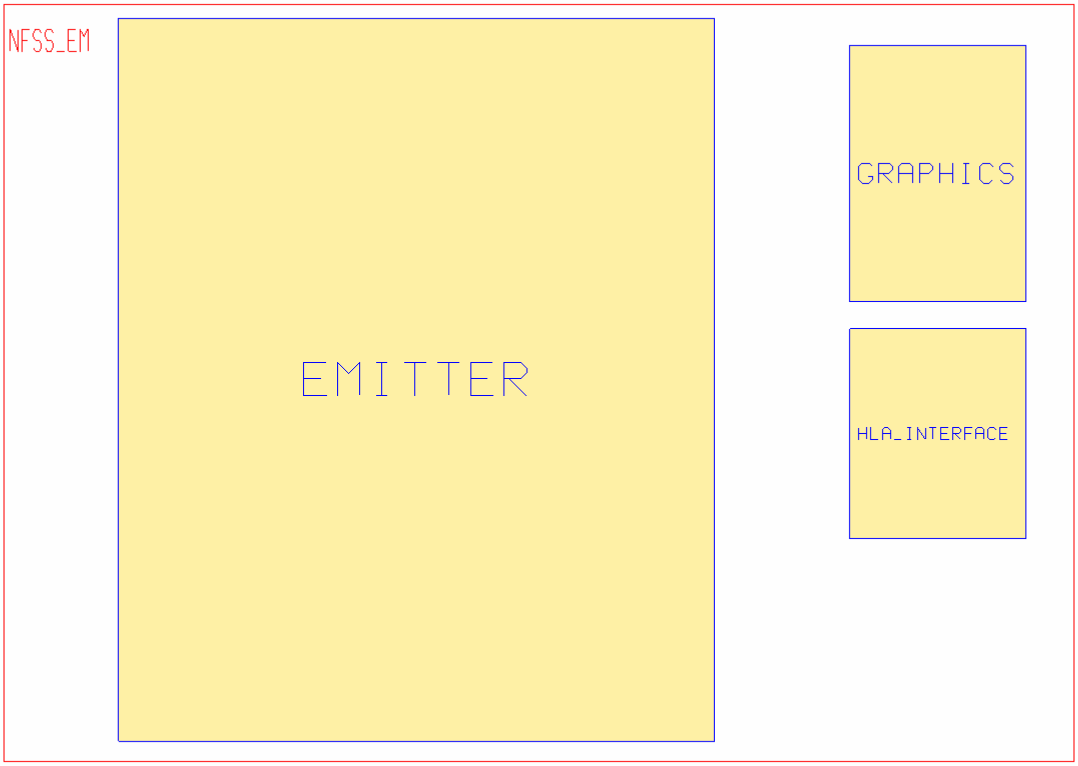


Figure 15 - Emitter Simulation Architecture

The name of this simulation is NFSS_EM. This fairly simple simulation consists of several models:

EMITTER: This elementary model defines the characteristics of the emitter being modeled. This model is considered elementary since it consists of processes and resources. A number of SFI input files are used to specify the characteristics of the emitter including things like frequency, bandwidth, modulation type, location, antenna height, transmit power, gain and polarization, transmit duration and intergeneration times, and waveform. Output files essentially capture the same information as the input files plus any interactive changes such as modified sensor locations and the interactive addition of new emitters.

GRAPHICS: This model handles interactive graphics events such as relocation and addition of emitter icons.

HLA_INTERFACE: This model "encapsulates" all of the handling of the HLA interface, and GSS makes the use of HLA very easy. This model is used to "publish" emitter information into the RTI Federation.

7.4.2 Unattended Ground Based (UGS) Sensor Simulation

The UGS models and simulation architecture drawing is shown in Figure 16.

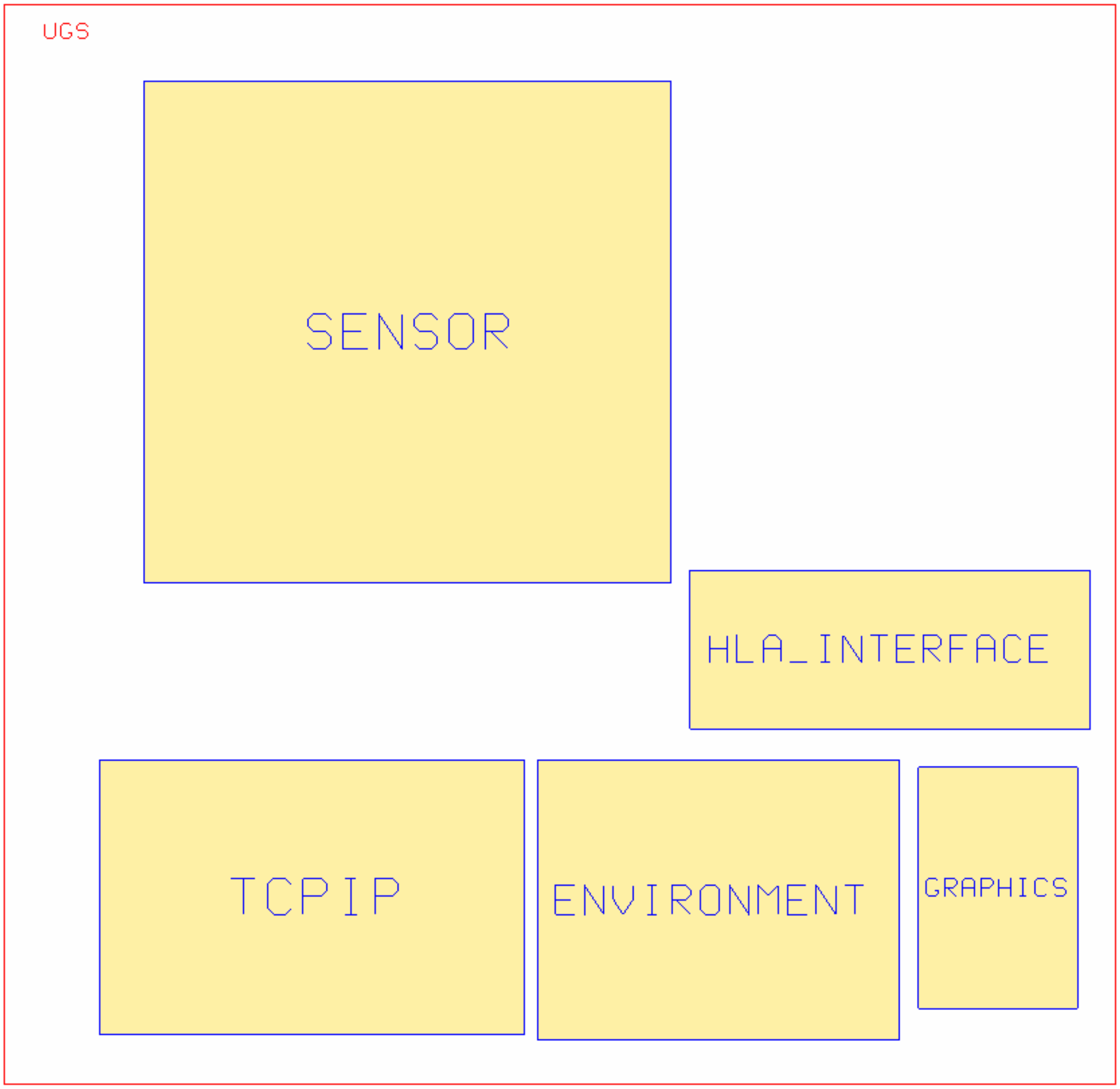


Figure 16 - Unattended Ground Based (UGS) Sensor Simulation Architecture

This simulation is named UGS. It consists of the following models:

SENSOR: This model models the characteristics of an unattended ground sensor. Input SFI files provide parameter values for antenna height, signal-to-noise threshold, receive gain, noise factor, location, and sensor action. It also handles movement or relocation of a sensor. Output files capture updates to sensor information. SENSOR provides for a user panel, i.e., dialog box, to view and modify attributes of a graphically selected sensor on the terrain.

TCPIP: This model is a TCP/IP client that will connect to the NFSS-OMS TCP/IP server port. It handles connection and set-up of a socket, and sending of information packets to the server. The SFI input file CONNECT.SFI specifies the IP address and port number of the server.

HLA_INTERFACE: This model "encapsulates" all of the handling of the HLA interface, and is used to "subscribe" to the information published by the emitter simulation.

ENVIRONMENT: This model handles the interface to PSI's Fast Propagation Prediction System (FPPS) library. It initializes the library, and makes calls to dynamically compute propagation loss between sensors and emitters taking into account effects of the 3D terrain data. FPPS has been optimized for speed and accuracy, and handles a range of frequencies from 20 MHz to 20 GHz.

GRAPHICS: This model handles interactive graphics events such as relocation, addition and selection of UGS icons.

7.4.3 NFSS-OMS Simulation

The NFSS OMS models and simulation architecture drawing is shown in Figure 17. NFSS-OMS is a fairly complex simulation since it provides a variety of functions. Whereas, the UGS and Sensor simulations both used just elementary models consisting of processes and resources, the NFSS-OMS uses a number hierarchical models that consist of models within models.

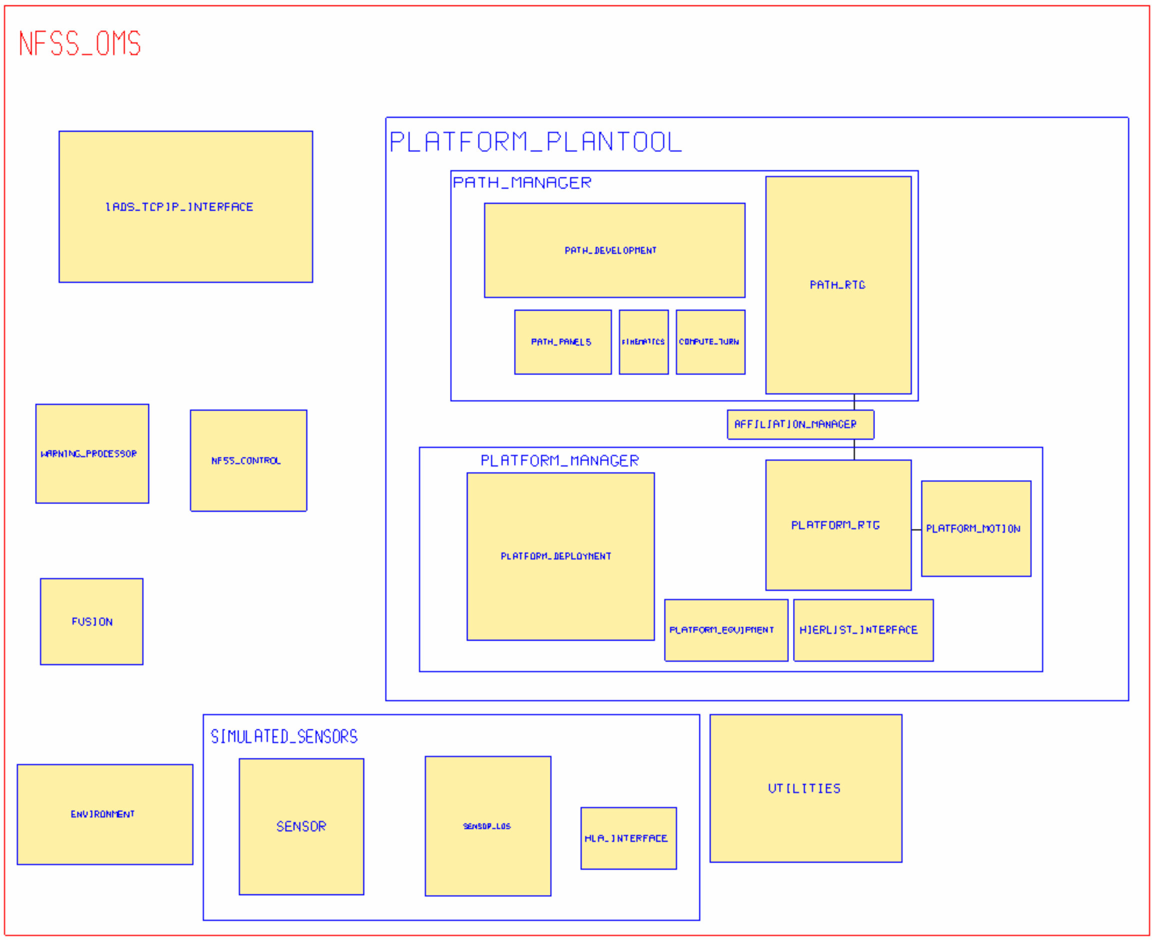


Figure 17 - NFSS-OMS Simulation Architecture

This simulation is named NFSS_OMS. It consists of the following models:

PLATFORM_PLANTOOL: This is hierarchical model consisting of two hierarchical models: PATH_MANAGER and PLATFORM_MANAGER. The first handles all the functionality to read-in, display, modify, add, and delete, etc. movement paths consisting of waypoints. It also provides display panels to view and modify waypoints, etc. PLATFORM_MANAGER manages platforms, which can be planes, ships, ground vehicles, etc. It can read in platform definitions from files that also specify the type of equipment that is loaded on each platform. This model also handles display of platforms, and provides display panels to modify platform equipment, etc. A smaller model named AFFILIATION_MANAGER is used to “join” the path and platform models and associates platforms with movement paths. In addition to supporting input from the UGS, the NFSS-OMS supports interactive addition of movement paths with airborne sensors. This capability uses the features of the models within this hierarchical model.

SIMULATED_SENSORS: This hierarchical model contains a Sensor model that is quite similar to the one used in the UGS simulation. It also contains a model for a line of sight (LOS) sensor system. In addition, the HLA interface model has been incorporated to this hierarchical model.

IADS_TCPIP_INTERFACE: This elementary model provides both a TCP/IP Client and Server. It provides user display panels to view and modify IP settings for both the client and server. An input file specifies the required IP address and port number. The server supports connections from sensor clients, and the client-side connects to the IADS CTP/IP server to report fused sensor data. See additional description under the section that follows on IADS.

WARNING_PROCESSOR: This model reads in parameters associated with threats such as frequency, bandwidth, waveform, entity information, alert-level, etc., and provides a panel to display when an sensor system output exceeds certain criteria and thresholds.

NFSS_CONTROL: This model handles graphics events, labels the programmable buttons in the GSS/RTG run time window, and handles connections to the UGS simulation.

FUSION: This model handles the sensor fusion calculations, and draws hierarchical “spot” reports of the fused data on the background terrain. It also provides a Spot Report Data panel to view data such as frequency, bandwidth, waveform, threat level, etc. for the emitters detected by the sensors. This model also draws the emitters on the terrain.

ENVIRONMENT: This model interfaces to the FPPS library to handle propagation calculations for different sensors, and computes and draws the coverage area of the sensors.

UTILITIES: Provides some functions used by the other models.

7.4.4 EBO-IADS Simulation

The model and simulation architecture drawing for IADS is shown in Figure 19. IADS is quite complex and contains models of many different types and fairly deep model hierarchies. The simulation is named IADS100. Figure 19 shows the IADS100 simulation drawing in a completely "uncovered" form - down to the individual processes and resources that constitute the models.

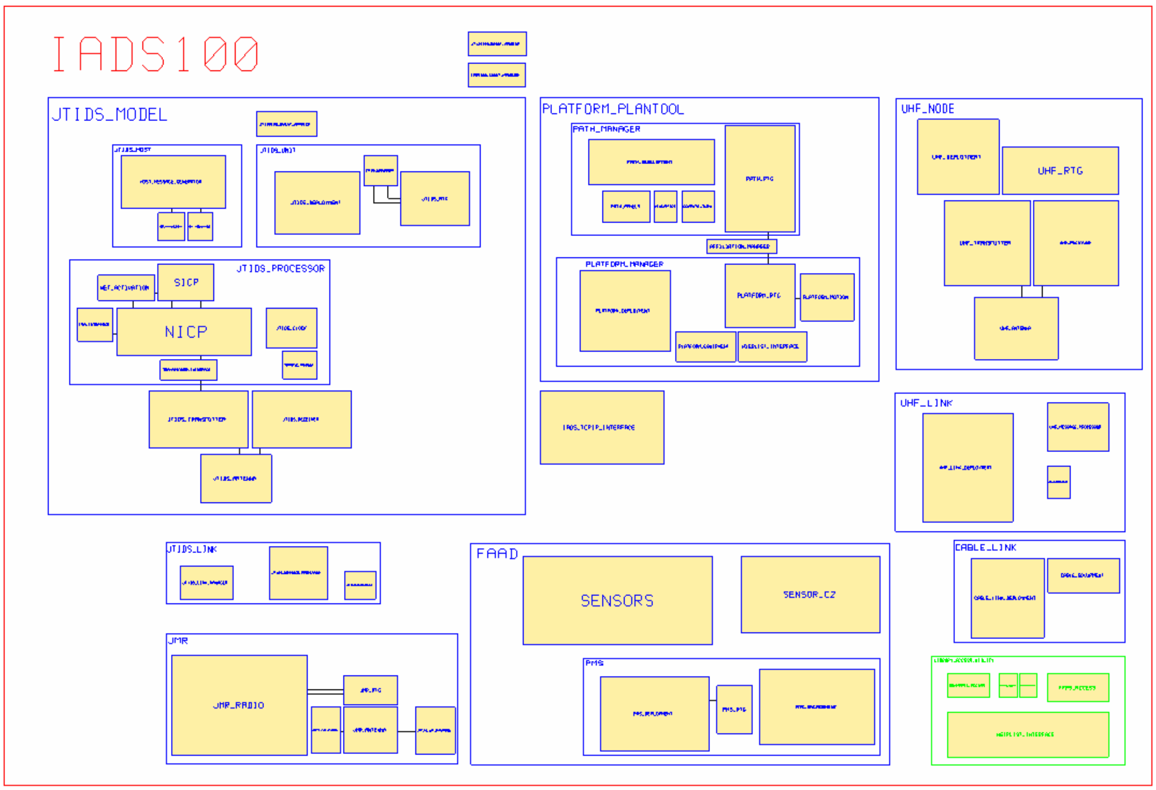


Figure 18 - Hierarchical Models of IADS

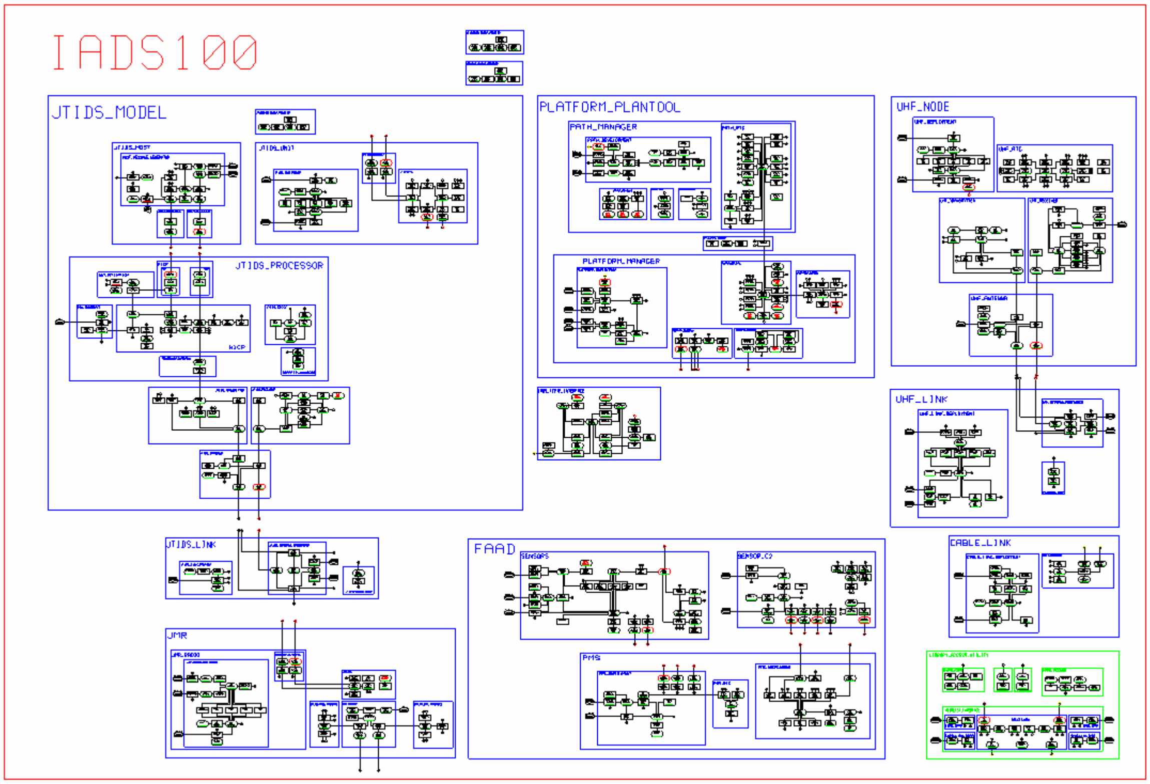


Figure 19 - IADS Detailed Simulation Architecture

There are several models used within IADS:

PLATFORM-PLANTOOL: This hierarchical model is essentially the same as the one used and discussed for the NFSS-OMS. It is quite common in PSI and with GSS to re-use models between simulations. This model handles the creation and modification of movement paths, e.g., flight paths, and for platforms like aircraft to be attached to movement paths and launched to follow the path. The platform manager sub-component handles the definition and modification of platforms and allows them to be equipped with radios, jammers, etc.

JTIDS_MODEL: This hierarchical model is quite complex and consists of several other hierarchical models. It is designed along the physical lines of an actual JTIDS radio.

- The JTIDS_HOST model handles message generation and input and output message queues.
- The JTIDS_PROCESSOR models the functions of the processor in the JTIDS radio, and includes models for Time Slot Assignment (TSA), traffic management, JTIDS clocking, the Subscriber Interface Computer Program (SICP), the transceiver interface, and Net Interface Computer Program (NICP).
- The JTIDS Transmitter and Receiver models model behavior of these respective JTIDS functions, and are connected to a model of the JTIDS Antenna.

- The JTIDS_UNIT model handles deployments of JTIDS radios based on input initialization files, maintains lists of JTIDS radios and other “bookkeeping” and handles part of the graphics interface.

JTIDS_LINK: This hierarchical model handles the links between JTIDS Radios, and includes models for a link manager, message processor, and message power.

UHF_NODE: This is a hierarchical model for UHF nodes, and includes models for a UHF transmitter, a UHF receiver, a UHF antenna, and a model to deploy nodes based on an initialization file. An additional model handles interactive graphics events such as inserts and removal of nodes, etc.

UHF_LINK: This is a hierarchical model of a UHF link, and includes a deployment model that initially deploys links based on an input file, a message processor which handles link connections and jamming, and a UHF message power model that computes path loss.

CABLE_LINK: The IADS simulation includes some cable connections in the ground networks. This hierarchical model simulates cable connections, and includes a model for cable link deployment that is driven by an input file, and a model of cable equipment.

JMR: This model handles computations associated with UHF jammers. The model supports both ground based and airborne jammers. IADS currently only uses airborne jammers. Input files determine the jammer radio characteristics.

FAAD: This is a fairly complex hierarchical model. FAAD stands for Forward Area Air Defense. Included in this model are models for Radar Sensor systems, Sensor C2 Units, and a model for a Pedestal Mounted Stinger (PMS) heat-seeking missile and a model of their associated Fire Unit. Each of these models uses input files that determine their characteristics, deployed locations and interconnectivity. Output files record changes to the scenario with respect to these models. The radar sensor model uses electromagnetic (EM) propagation calculations to determine if aircraft is in range, and takes into account effects of terrain masking. The radar sensor model tracks and coasts targets and reports the air picture to the Sensor C2 Unit. The command and control (C2) model makes a fire determination and communicates with the fire units (with PMS instances). The Radar, C2 and Fire Unit models all support interactive events that can include additions, movements, deletions, and activation/deactivation events. New units can be added, and existing units can be repositioned. The tracking radar for the Fire Units can be dynamically repositioned, and it uses EM propagation calculations to pick out and track targets.

IADS_TCPIP_INTERFACE: The model is essentially the same as the one used for TCP/IP in NFSS-OMS. The visual development environment of GSS allows a user to zoom into a drawing to see more detail. Figure 20 shows a close-up of this model, and one can clearly see the client components on the left and the server components on the right. Resources associated with user display panels are labeled in red. GSS models are frequently reused across simulations. Within IADS, only the server functionality is used to support a connection from the NFSS-OMS client to report emitters detected by the sensors.

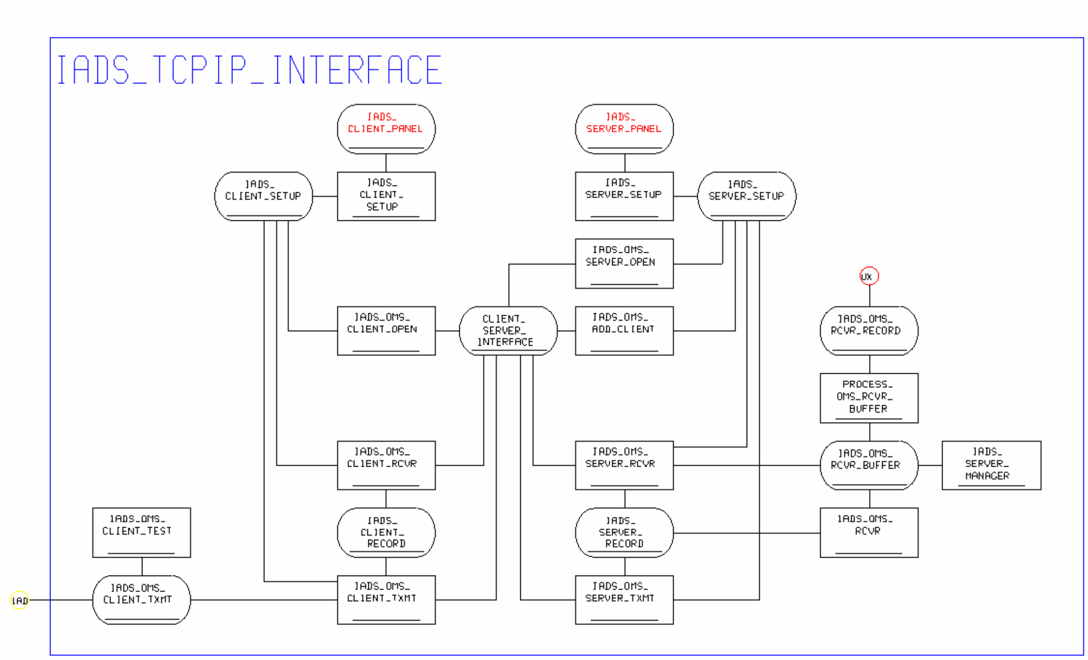
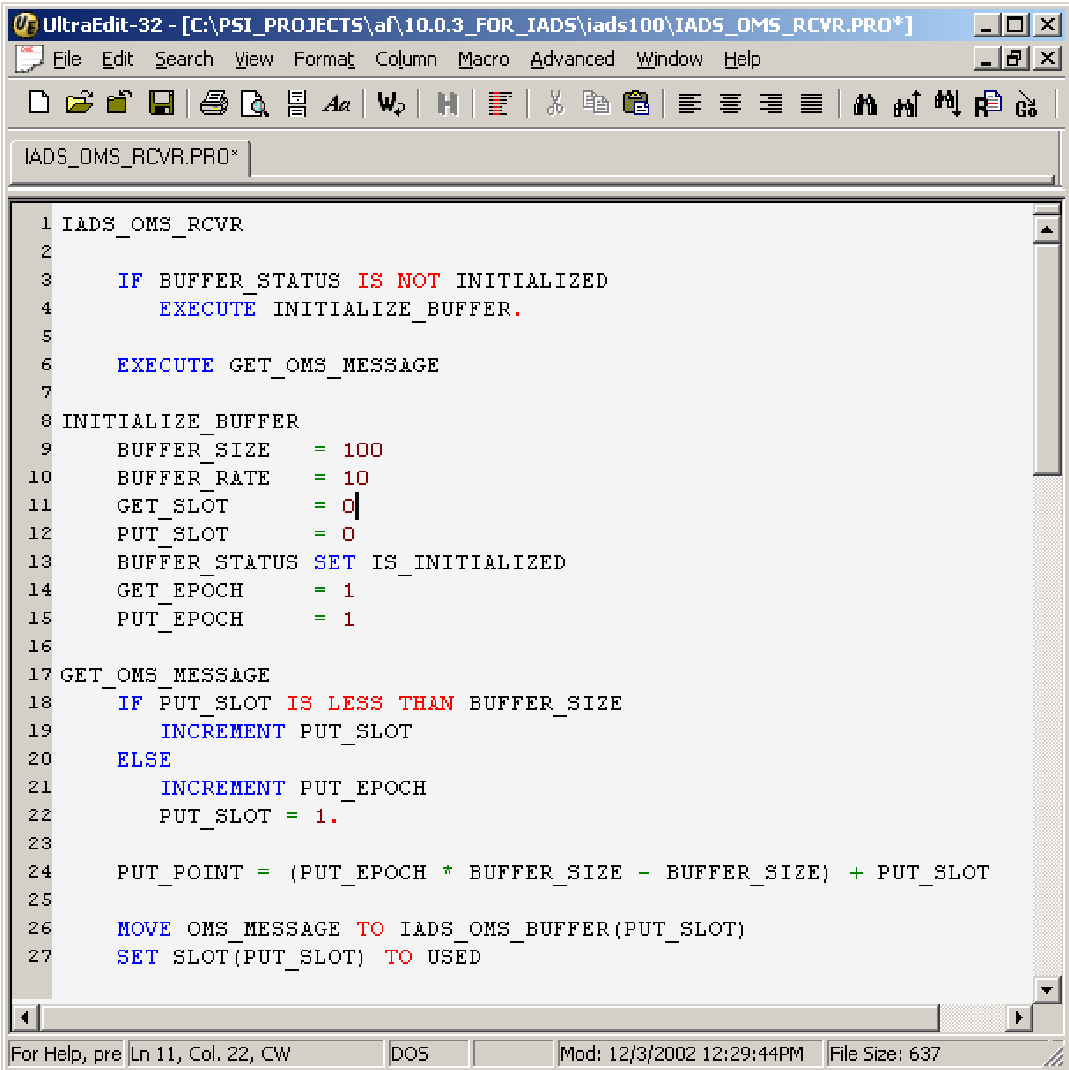


Figure 20 - IADS_TCPIP_INTERFACE Model

As mentioned a couple of times in the earlier sections of this report, the hierarchical modeling capabilities of GSS provide a direct connection between high-level architecture and the underlying rules that determine behavior.

If you double click on a process, it opens that process in an editor window to view and modify the rules for that process. Model resources are viewed and edited in a similar manner. Figure 21 shows a screen shot of the editor window for the IADS_OMS_RCVR process taken from a Windows platform. As you can see, the GSS process language is a very high-level, English-like language. Key words in GSS are color coded to improve understanding.



```
IADS_OMS_RCVR

    IF BUFFER_STATUS IS NOT INITIALIZED
       EXECUTE INITIALIZE_BUFFER.

    EXECUTE GET_OMS_MESSAGE

INITIALIZE_BUFFER
    BUFFER_SIZE    = 100
    BUFFER_RATE    = 10
    GET_SLOT       = 0
    PUT_SLOT       = 0
    BUFFER_STATUS SET IS_INITIALIZED
    GET_EPOCH      = 1
    PUT_EPOCH      = 1

GET_OMS_MESSAGE
    IF PUT_SLOT IS LESS THAN BUFFER_SIZE
       INCREMENT PUT_SLOT
    ELSE
       INCREMENT PUT_EPOCH
       PUT_SLOT = 1.

    PUT_POINT = (PUT_EPOCH * BUFFER_SIZE - BUFFER_SIZE) + PUT_SLOT

    MOVE OMS_MESSAGE TO IADS_OMS_BUFFER(PUT_SLOT)
    SET SLOT(PUT_SLOT) TO USED
```

Figure 21 - IADS_OMS_RCVR Process Rules

7.4.5 SUMMARY

This section of the Final Report presented the models and simulation architectures for the four simulations used to built the GIESim demonstration. It showed the architectural drawing for each simulation, and listed the models contained along with their purpose. The ability of GSS to zoom into any area of the drawing was demonstrated, e.g., IADS-TCPIP_INTERFACE, and an example of the GSS process language was given. The GSS hierarchical approach to modeling along physical lines was described along with some discussion of model reuse. The intent of this section was to show the modeling and architectural approach used with GSS to illustrate some of the attributes of GSS discussed in the GSS Attributes section of this report.

8. RESEARCH PROPOSAL FOR 2003

The final work item completed in the first round of the GIESim effort by PSI was the development of a PSI research proposal for 2003.

- Definition of critical interface requirements associated with an Intelligent Simulation Interface framework.
- Research requirements of applications in need of real solutions, and benchmark alternative solutions.
- Selectively choose applications to potentially support under GIESim.
- Test ideas, processes, and concepts against actual application needs.
- Leverage existing M&S work to demonstrate GIESim in the near-term.
- Research large-scale simulation frameworks to understand their advantages and limitations to leverage and help guide the GIESim effort and evolution.
- Provide a way to advertise emergent GIESim capabilities and direction through early contact with potential applications.

9. SUMMARY

In the first round of efforts on the evolution and development of the GIESim Lab, PSI helped to launch the GIESim Lab through analysis, reports, and demonstrations. PSI is committed to help make the realization of the GIESim Lab a success. PSI has produced several inputs to the GIESim lab including ideas and requirements analysis for a successful lab, attribute lists for excellent M&S, model taxonomy, and simulations for the GIESim lab. We will continue to provide ideas, material, simulations and planning tools to ensure it's success. We produced a multi-computer simulation to demonstrate the kinds of large-scale, complex simulations that GIESim is aimed at. We also produced a research proposal for 2003 aimed at ensuring the success of the emergent GIESim capabilities. We look forward to working with the other members of the team and the other tools to provide the best solution for the Air Force in future rounds of work on GIESim.

10. REFERENCES

[1] GIESim Overview Material from AFRL, Rome Research Site

[2] Materials from the May 22-33 Kick-off meeting, AFRL, Rome Research Site

[3] White Paper and Briefing on Information for Global Reach (IFGR), AFLR, Rome

[4] The GSS Run-Time Graphics (RTG) Reference Manual, PSI, Spring Lake

[5] Building Reusable Models Using GSS, PSI, Spring Lake

[6] GIESim Kick-off Follow-up Notes, J. Periard, SRC

[7] Technical Attributes of M&S [Report #3]

[8] Business Attributes of M&S [Report #3]

[9] Preliminary GSS Model Taxonomy [Report #3]

ATTACHMENT I – FINAL 12/04/02 GIESIM BRIEFING SLIDES

Slide 1

Slide 2

Slide 3

GIESim Topics

- toward achieving success

- Potential users / application requirements
- Generic infrastructure requirements
- Generic approach to model architectures
- Generic approach to simulation to:
  - cut the time
  - cut the cost

Slide 4

GIESim Topics

- toward achieving success

***Generic* implies:**

- Independent of the tools used
- Independent of the specific implementation

Slide 5

GIESim

POTENTIAL USERS / APPLICATIONS

GIESim   PREDICTION SYSTEMS, INC.

Page 5

Slide 6

GIESim - Potential Users / Applications

Identifying the Market

PROGRAMS:

Air Force

- **GSTF** - AF/XI (AT AOC, DTIG, MC2C) , PBA, EBO, JBI, etc..
- **AF Joint Synthetic Battlespace**
- **TCT/TUT** - GMTI, AMSTE, DCGS
- **Global Mobility** - Smart Tankers , IFGR, GATM
- **ISR** - NCCT ACTD, Data Links

USN - CEC

USA - WIN-T, FCS

DoD - NetWars, GIG

Homeland Security

GIESim   PREDICTION SYSTEMS, INC.

Page 6

Slide 7

GIESim - Potential Users / Applications

Identifying the Market

POTENTIAL USERS:

Air Force

- AF/XI/XC (AC2ISR, ESC, AFCA, …)
- ACC, AMC, PACAF, AFSPC

DoD

- DARPA, OSD/C3I, DISA, USJFCOM, USSOCOM, USSPC

USN - SPAWAR

USA - CECOM (I2WD, C2D, …)

Homeland Defense

- NorthCOM , USSPC , PACOM, FEMA, NIPC

GIESim PREDICTION SYSTEMS, INC.

Slide 8

GIESim - Potential Users / Applications

Identify types of support:

INVESTIGATIONS

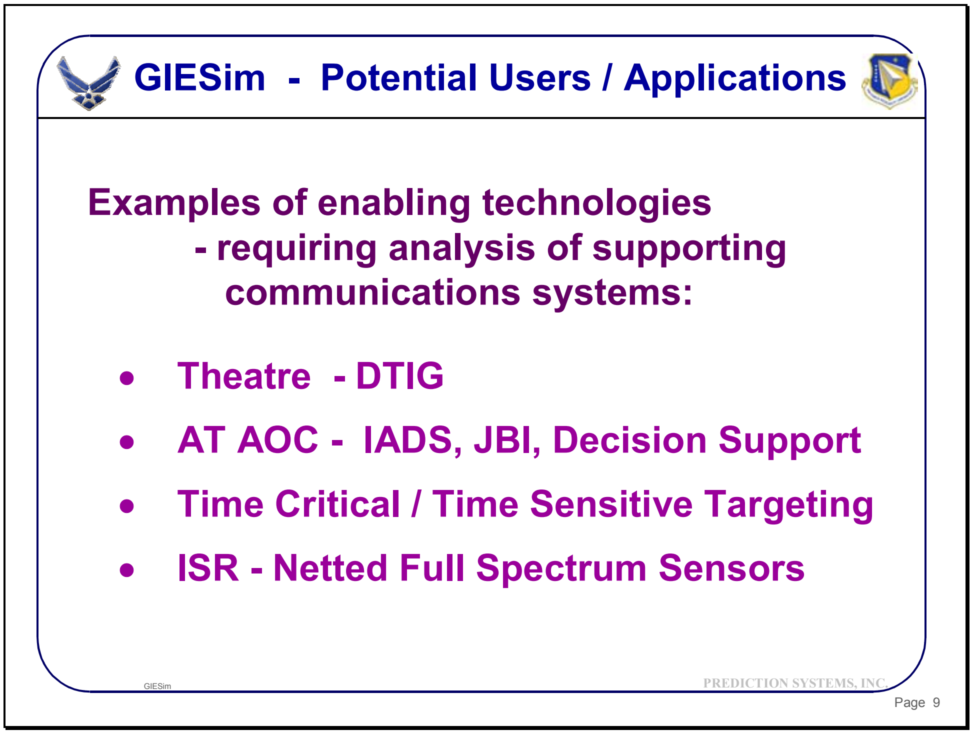
PROGRAM STAGES

APPLICATIONS 11/12/02

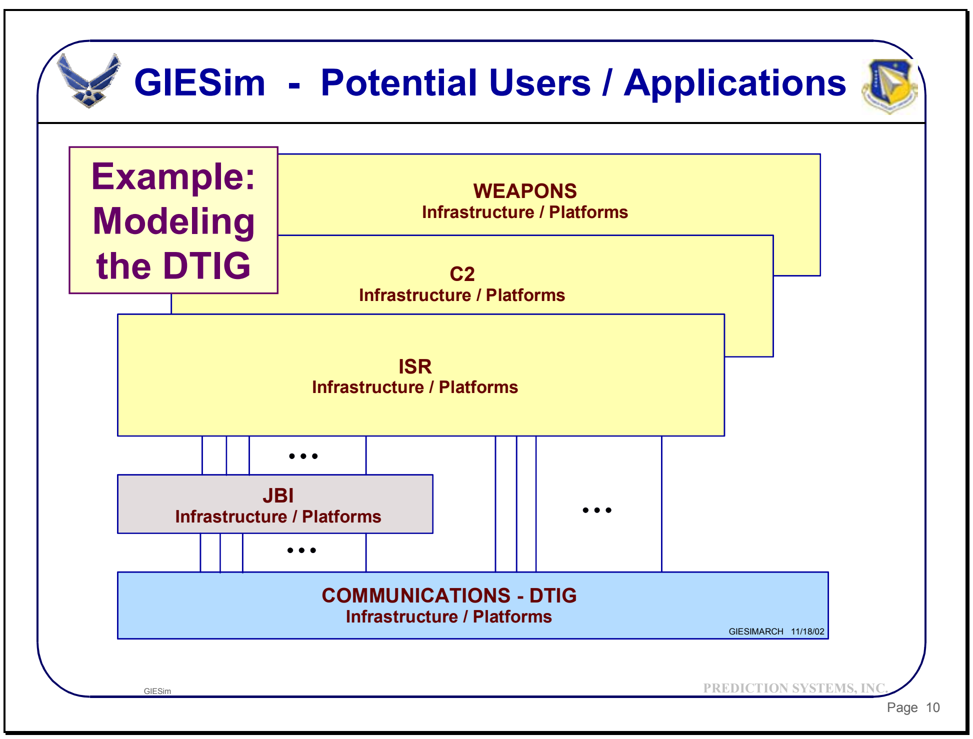
	Connectivity	Capacity	Assurance	•••
Requirements Analysis	x	x	x	•••
System Analysis	x	x	x	
System Design	x	x	x	
System Development		x		
System Test	x	x	x	•••

GIESim PREDICTION SYSTEMS, INC.

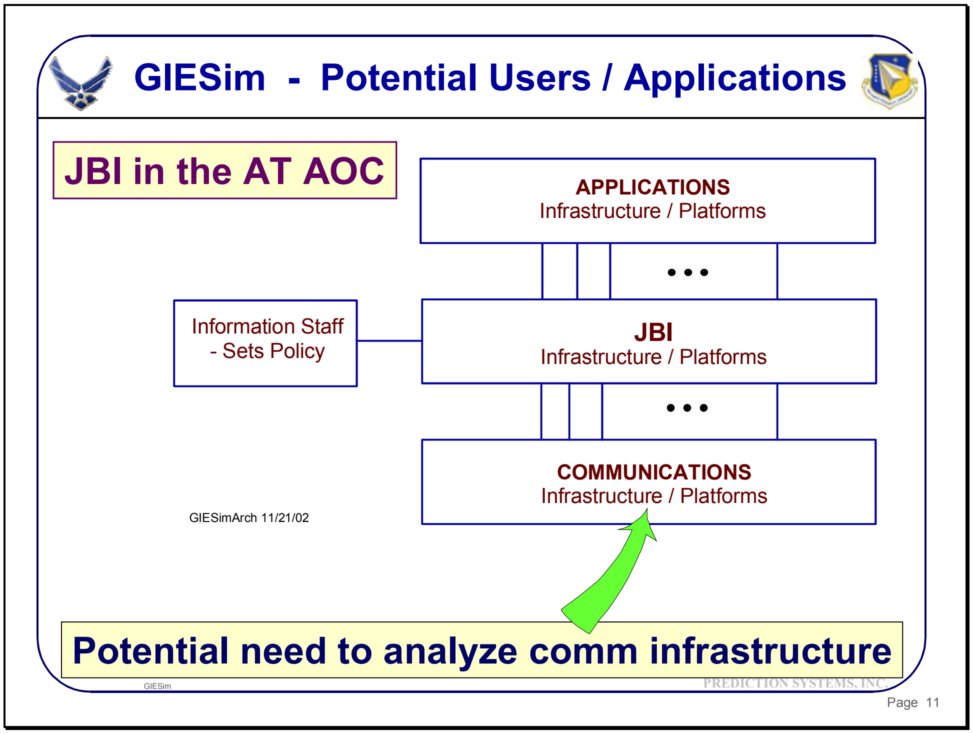
Slide 9



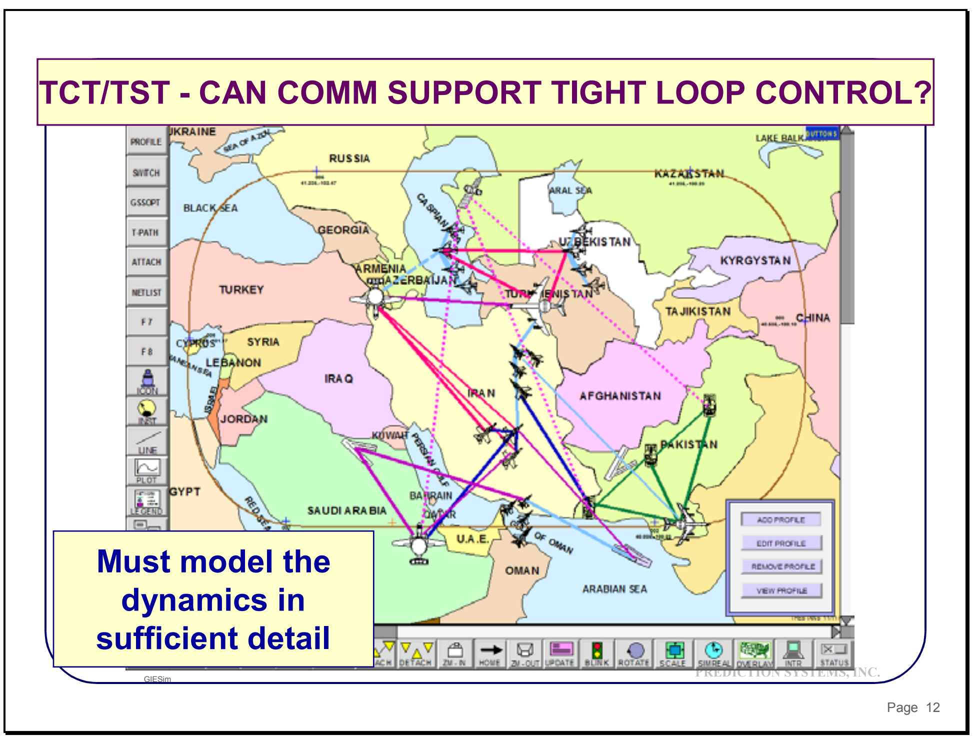
Slide 10



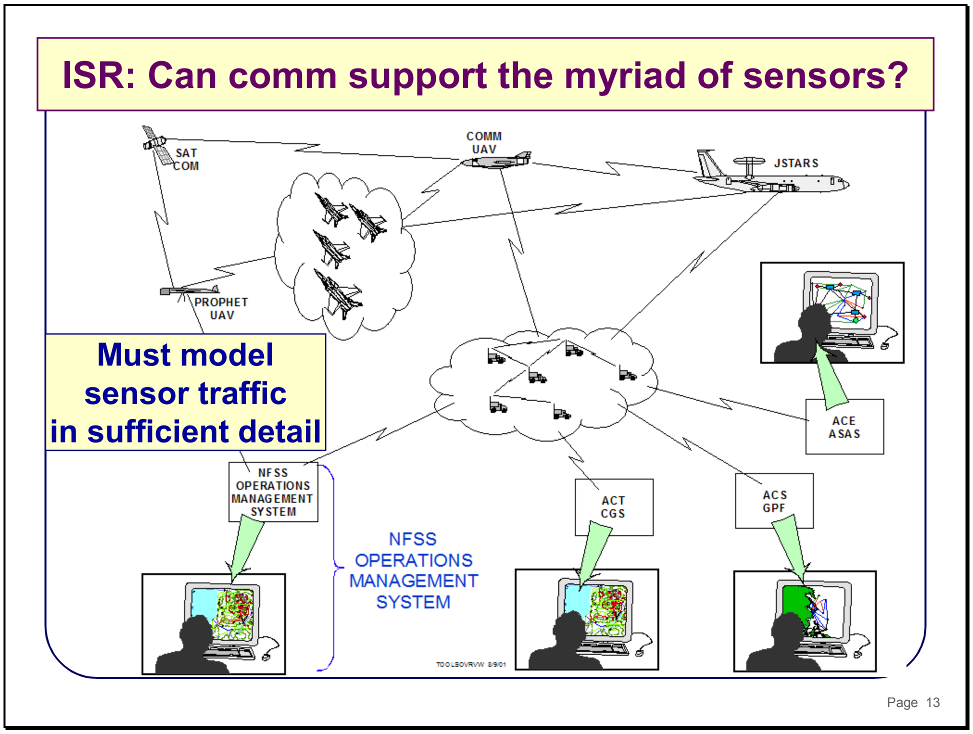
Slide 11



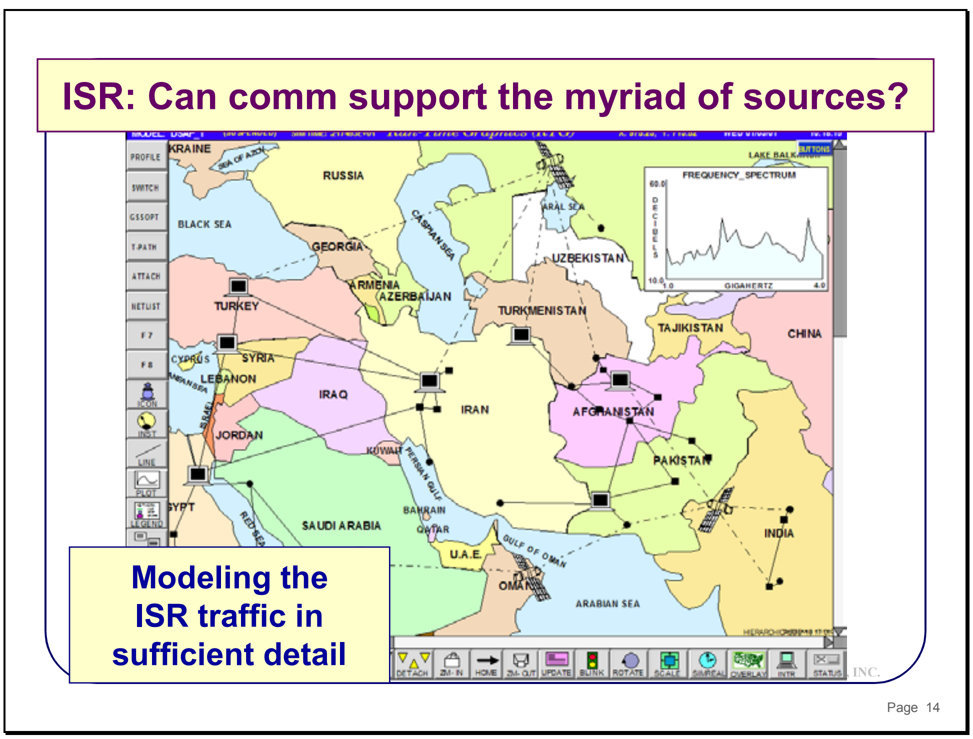
Slide 12



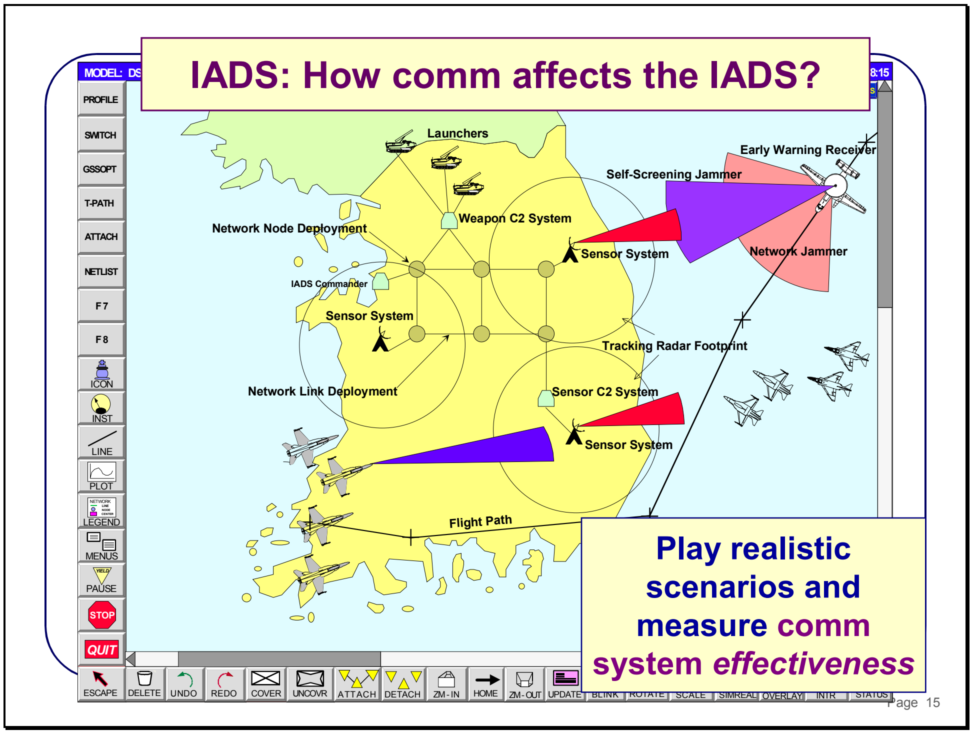
Slide 13



Slide 14



Slide 15



Slide 16

GIESim

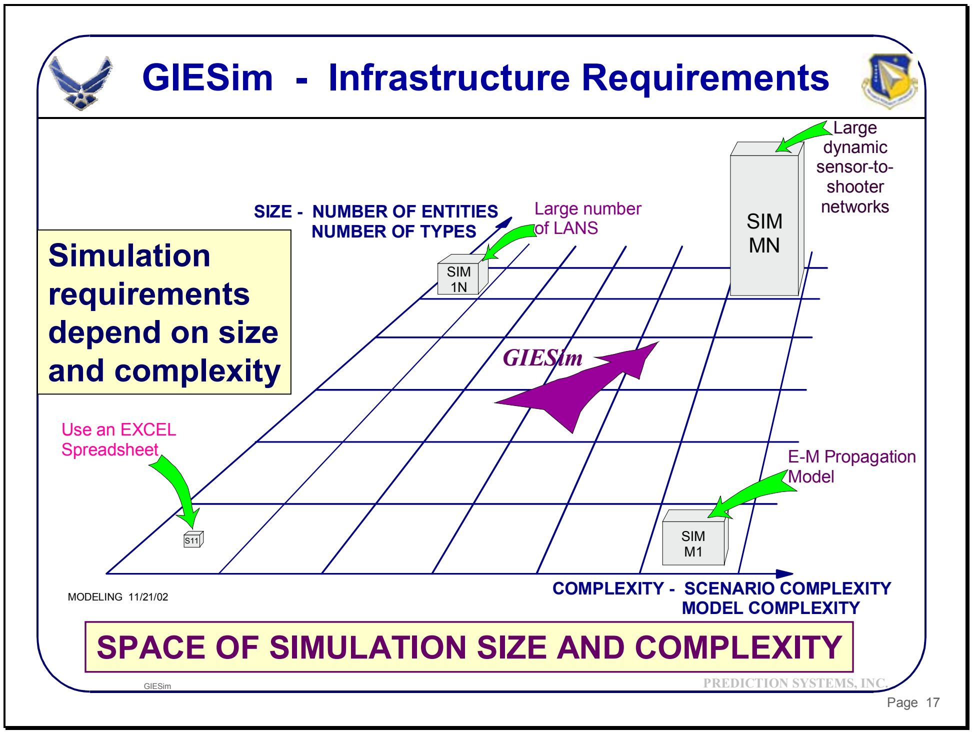
GENERIC INFRASTRUCTURE REQUIREMENTS

GIESim

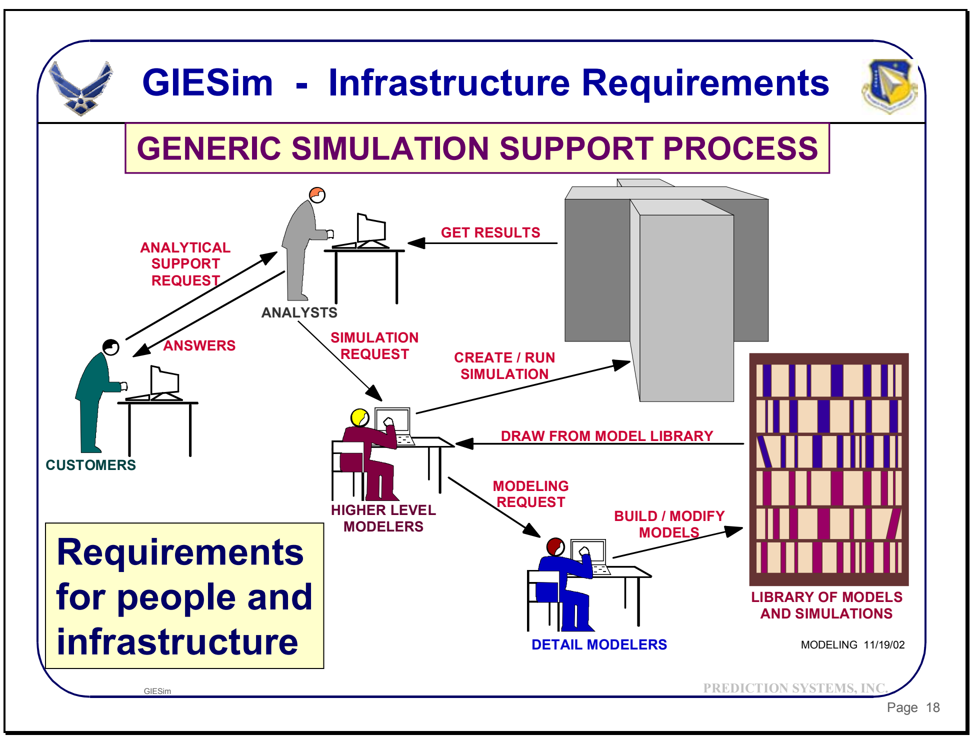
PREDICTION SYSTEMS, INC.

Page 16

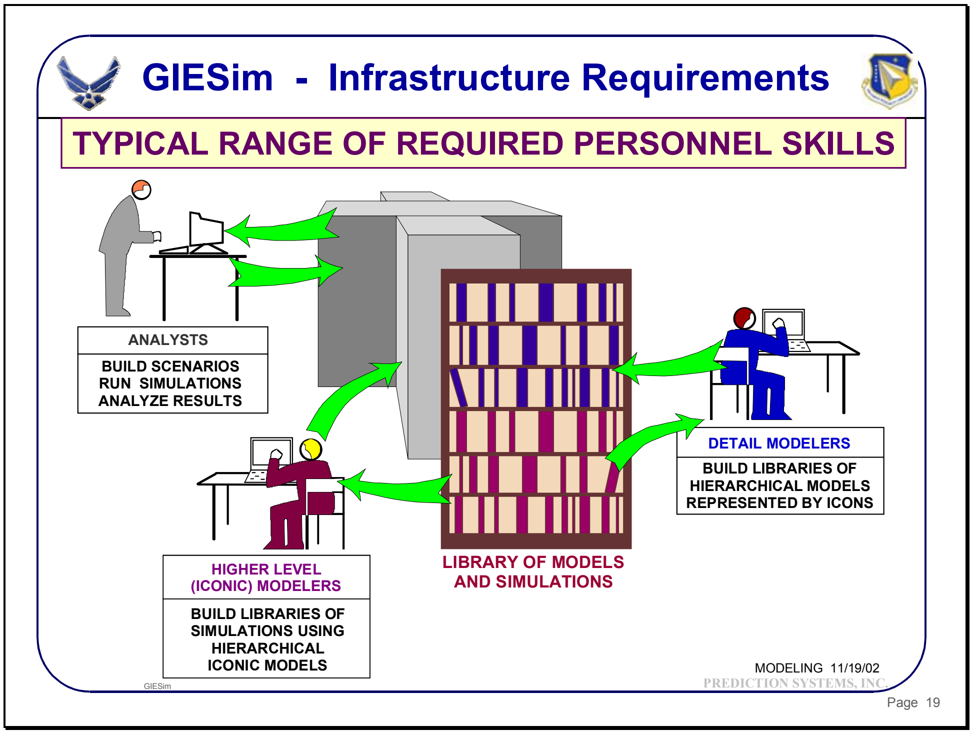
Slide 17



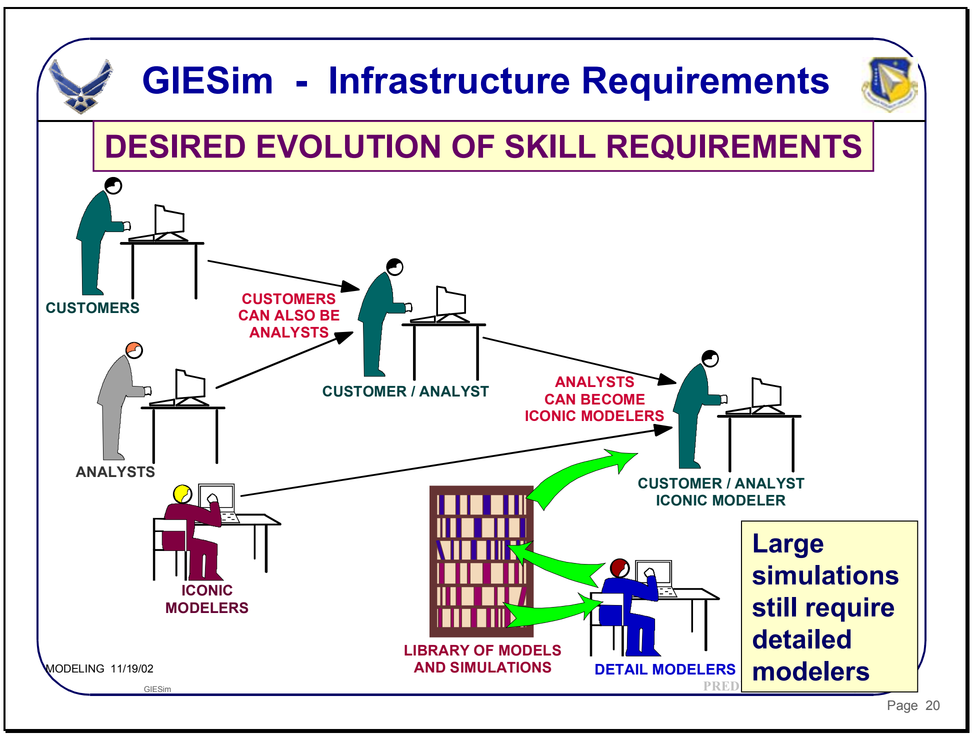
Slide 18



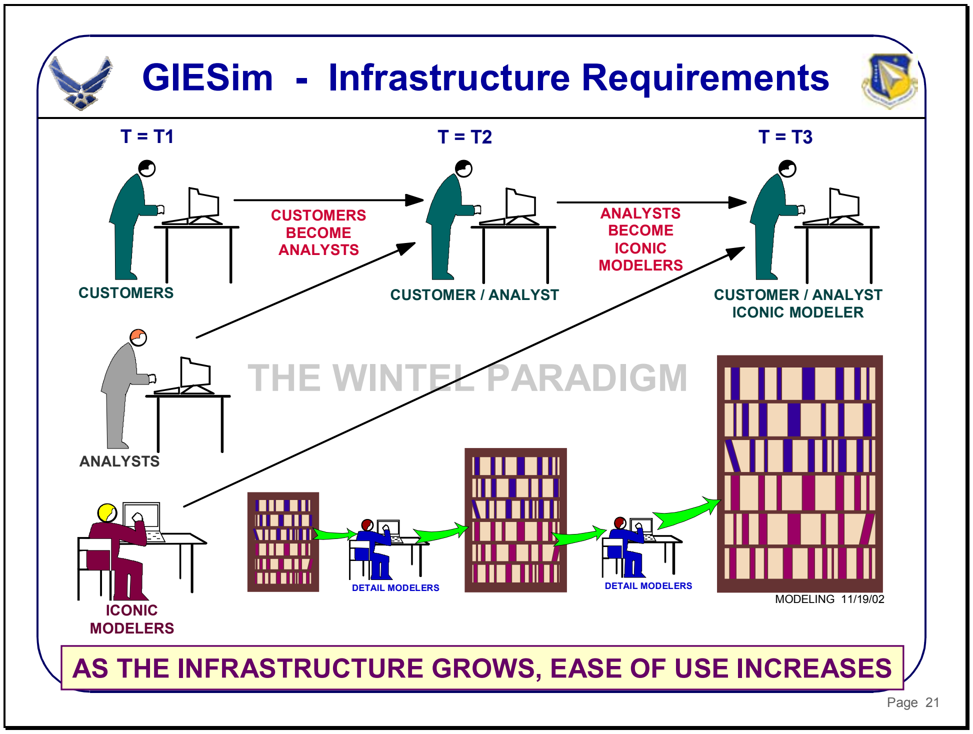
Slide 19



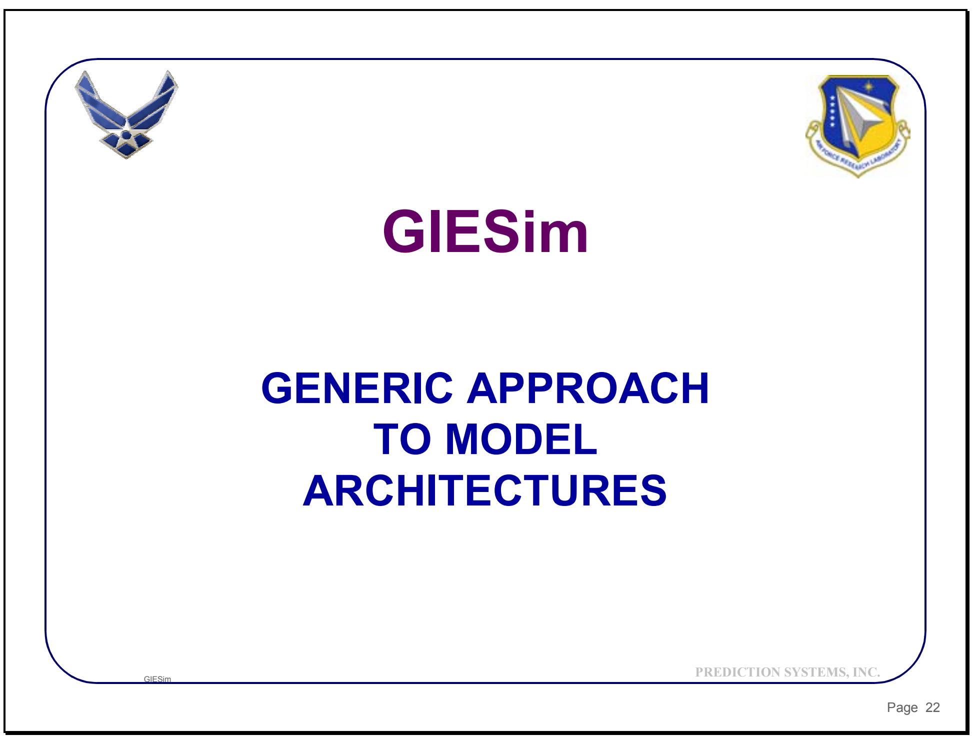
Slide 20



Slide 21



Slide 22



Slide 23

GIESim - Model Architecture Requirements

To support different analysis using GIESim, models must be provided
- rapidly & at low cost

- **The model architectures are critical to achieving this goal.**
- **The model architectures can be designed independent of the tools used to build them.**
- **They must support hierarchical iconic modeling.**

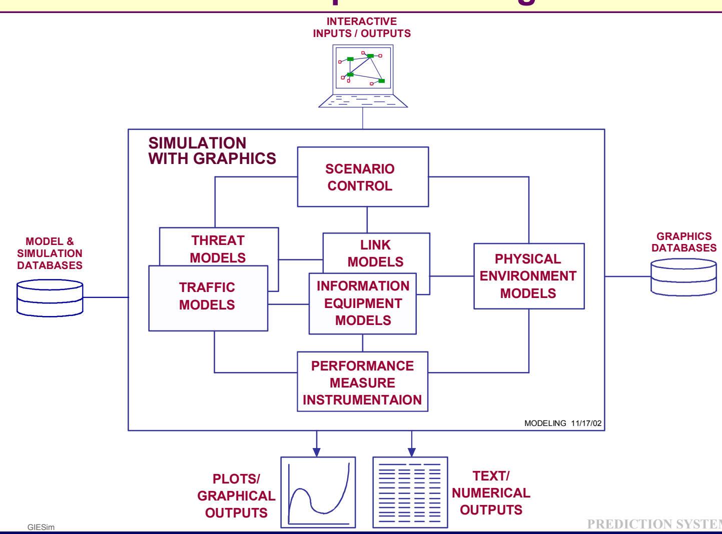
GIESim PREDICTION SYSTEMS, INC

Page 23

Slide 24

GIESim - Model Architecture Requirements

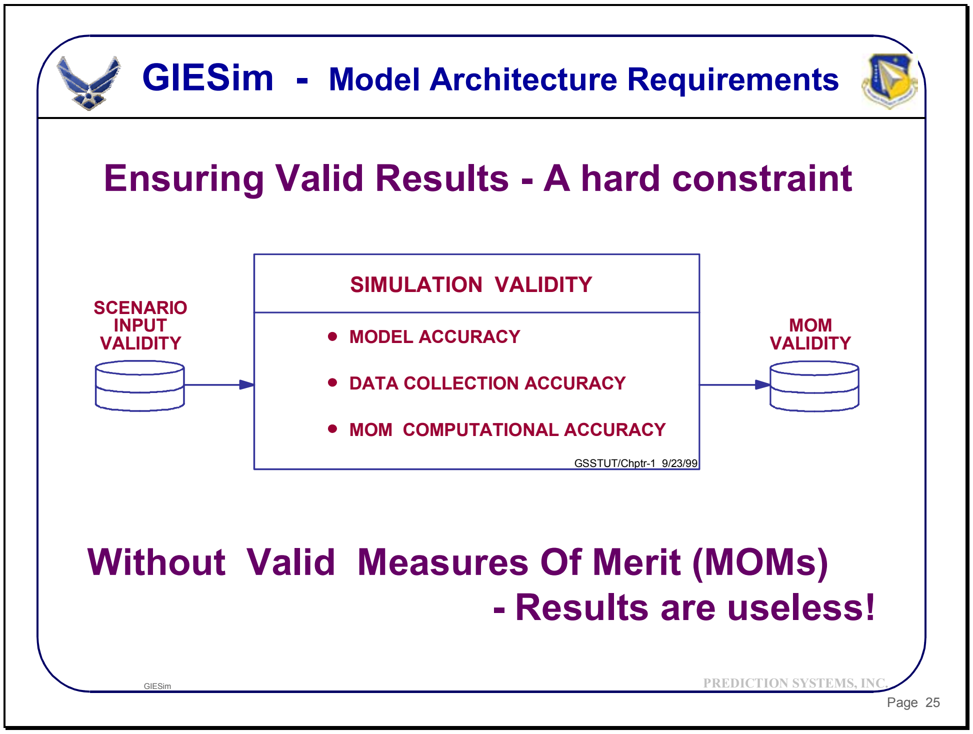
Model architectures - planned to get results fast



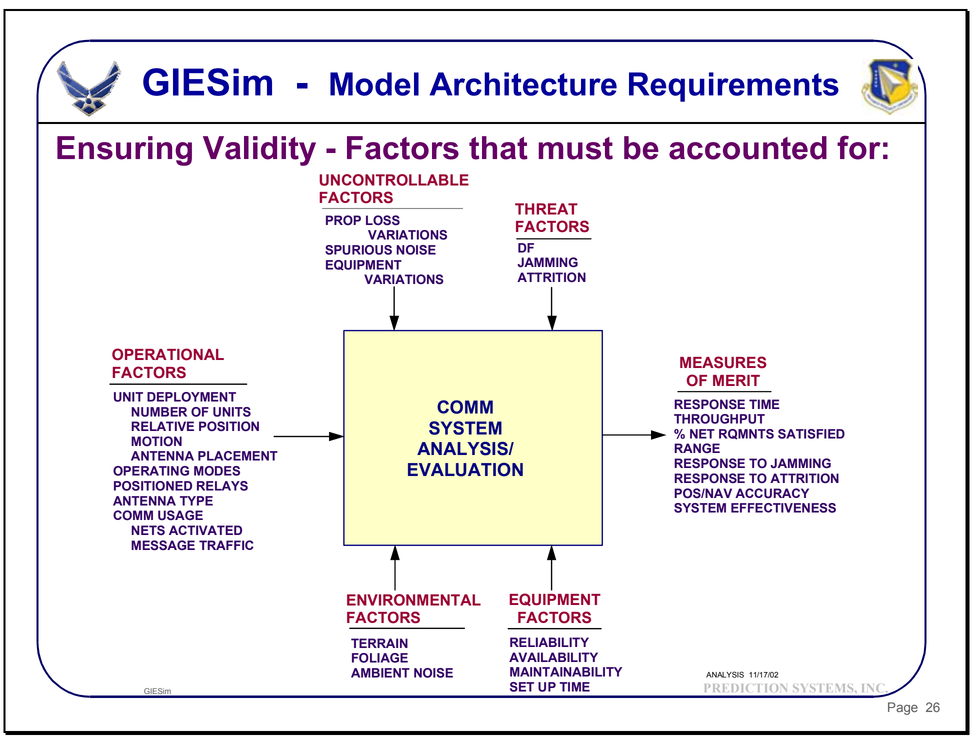
GIESim PREDICTION SYSTEMS, INC

Page 24

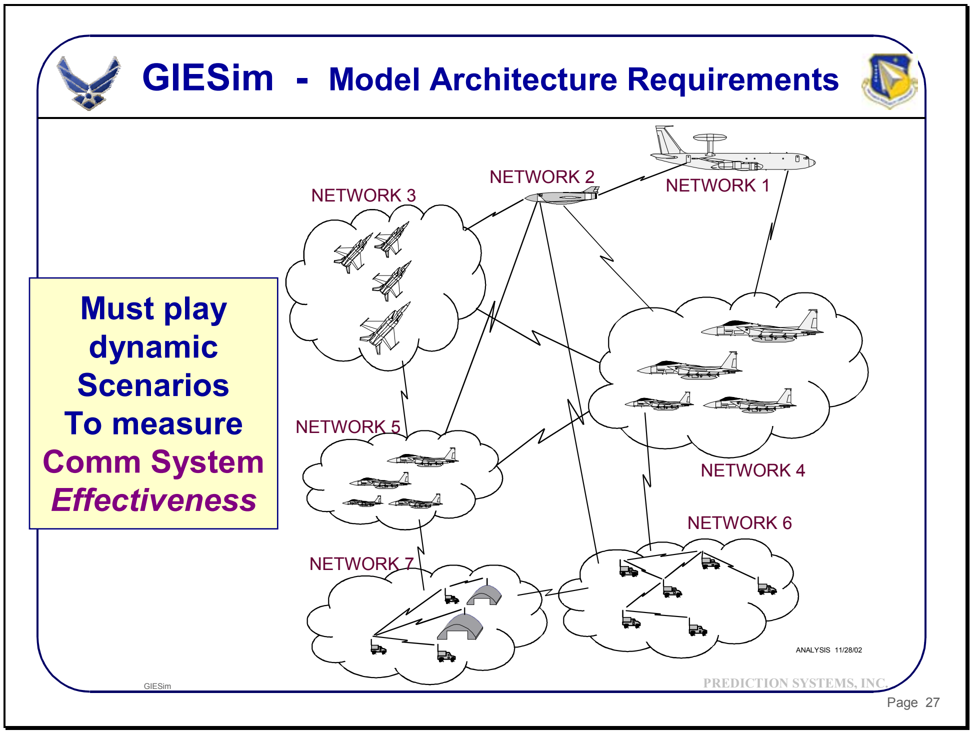
Slide 25



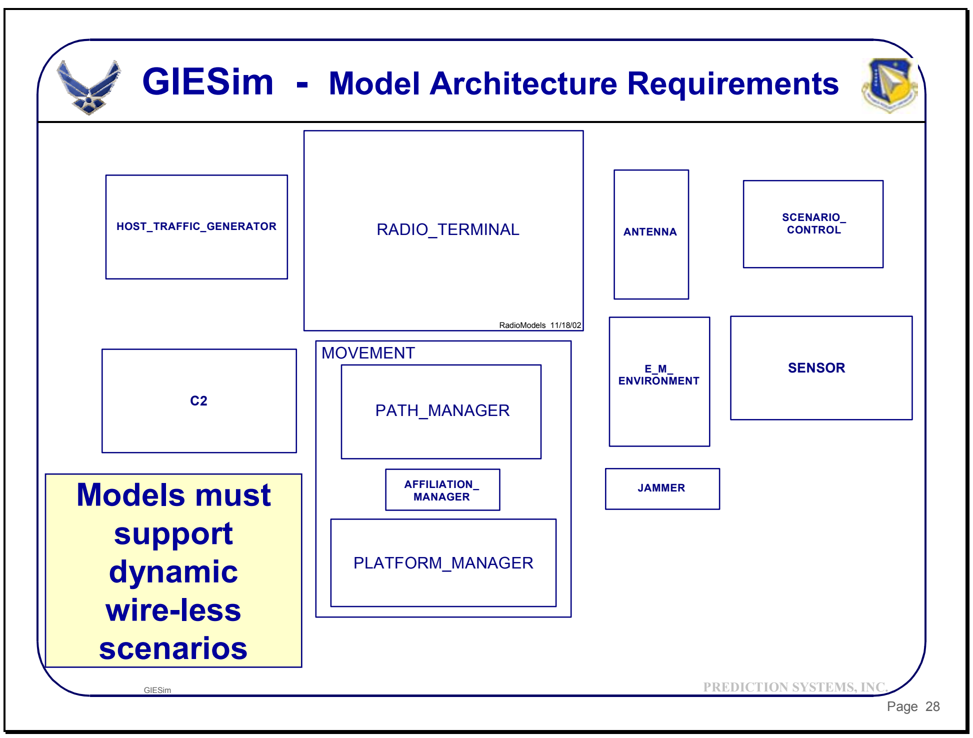
Slide 26



Slide 27

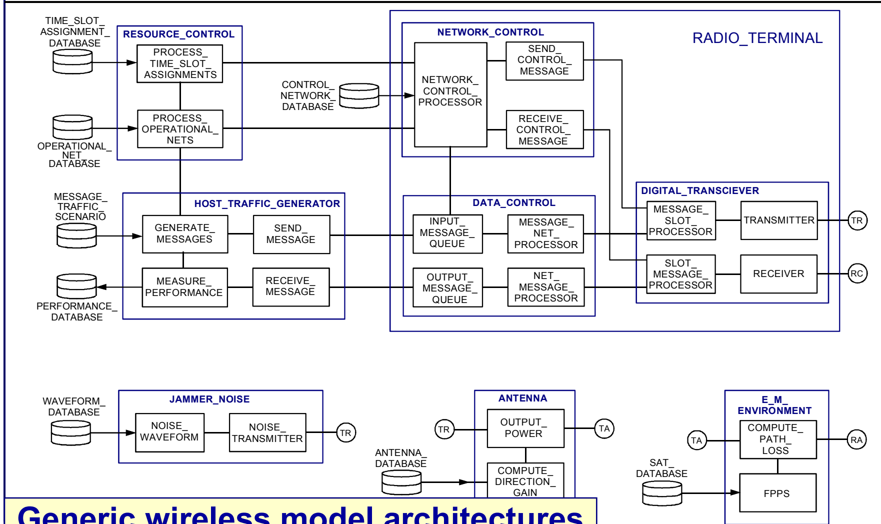


Slide 28



Slide 29

GIESim - Model Architecture Requirements



Generic wireless model architectures can be tailored for specific equipment

Slide 30

GIESim

Generic approach to simulation to:
- cut the time
- cut the cost

Slide 31

GIESim Objective Capability:

Provide *Valid* Modeling & Simulation

Solutions in *days / weeks*

(instead of months / years)

that:
- **run fast**
- **are easy to use**
- **support informed decisions**

& Save precious *time* and *$$*

GIESim

PREDICTION SYSTEMS, INC.

Page 31

Slide 32

GIESim - Generic Approach

To accomplish these goals, must be able to:

- **Create tailored simulations - *fast***
  - **from a library of models**
  - **using people who are *not* experts**
- **Create complex scenarios - *fast***
  - **using interactive graphics**
  - **with people who are *not* experts**
- **Add new models to the shelf - *fast***
  - **using interactive graphics**
  - **need subject-area experts**

GIESim

PREDICTION SYSTEMS, INC.

Page 32

Slide 33

GIESim - Generic Approach

The critical pieces are:

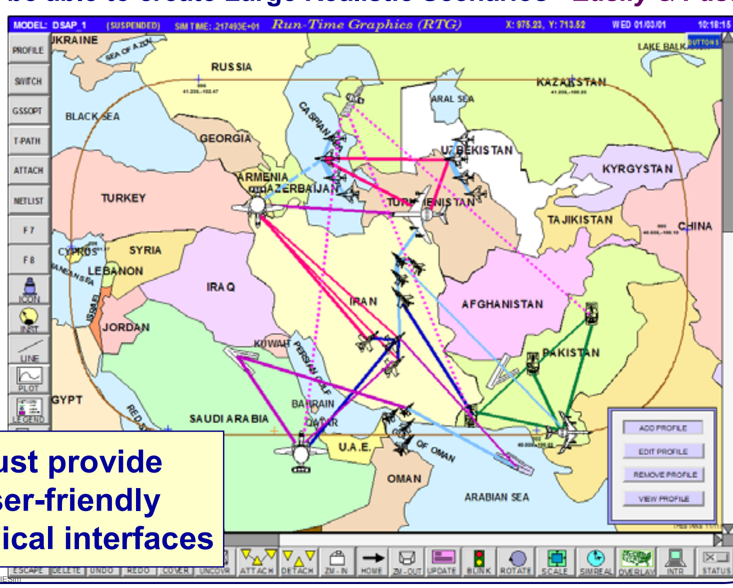
- **Existing simulations that provide**
  - **A basis for client exposure / acquisition**
- **A shelf of models to draw upon**
  - **Switches, routers, radios, etc.**
  - **Platforms, movement, etc.**
  - **E-M-Environment (terrain, foliage, etc.)**
- **A model architecture framework that is integrated / interoperable & supports rapid:**
  - **Development / modification of models**
  - **Incorporation of models into simulations**

GIESim PREDICTION SYSTEMS, INC.

Page 33

Slide 34

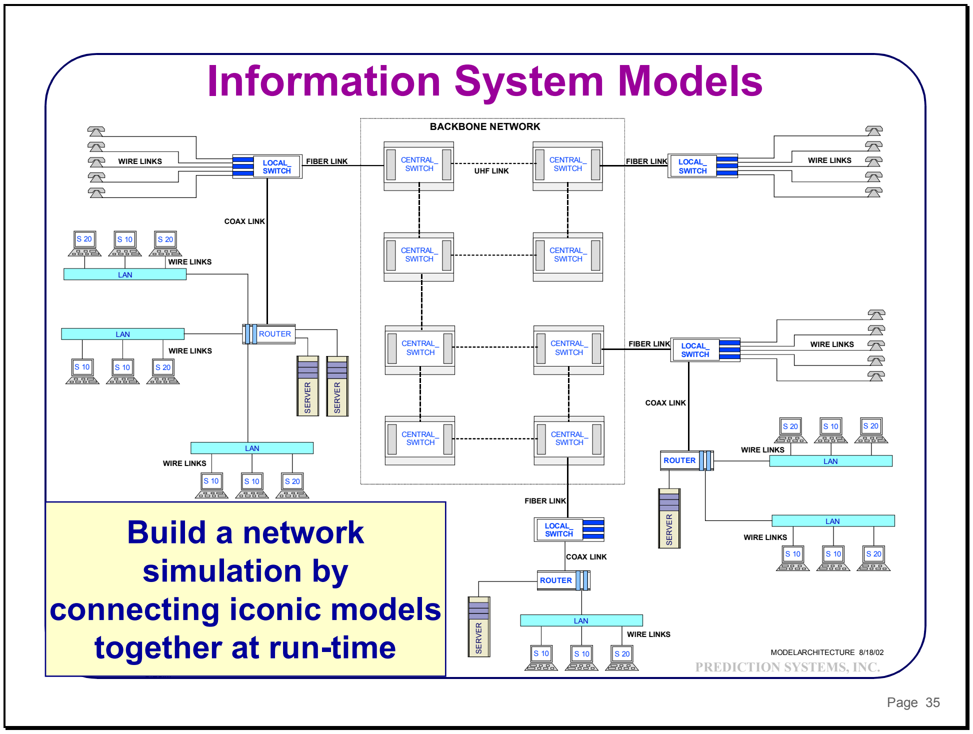
Must be able to create Large Realistic Scenarios - *Easily & Fast*



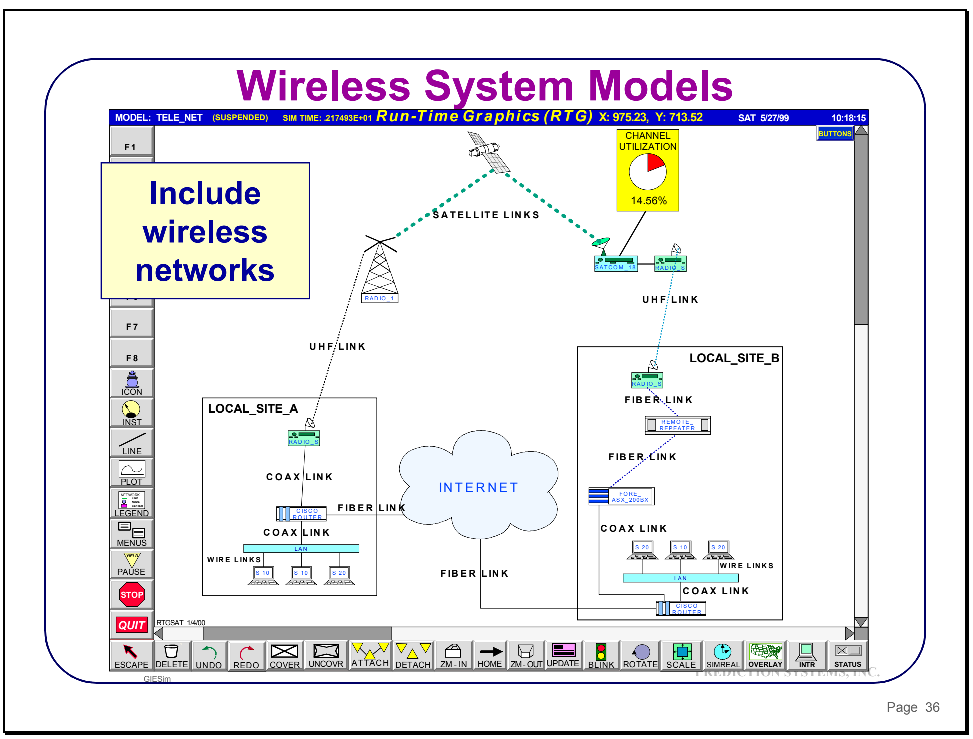
Must provide user-friendly graphical interfaces

Page 34

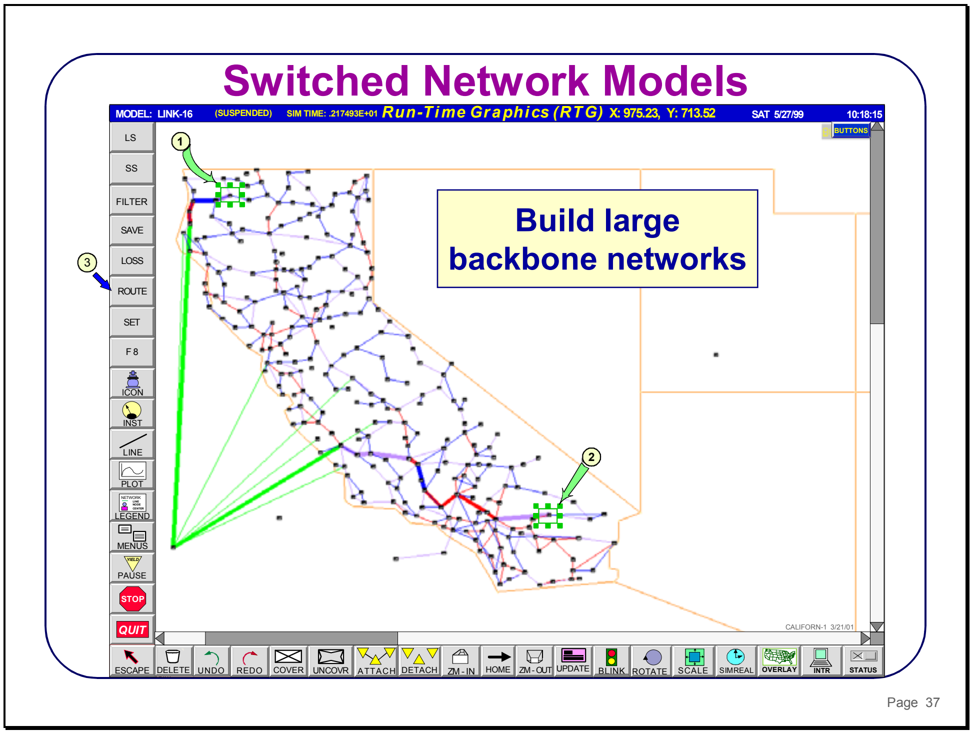
Slide 35



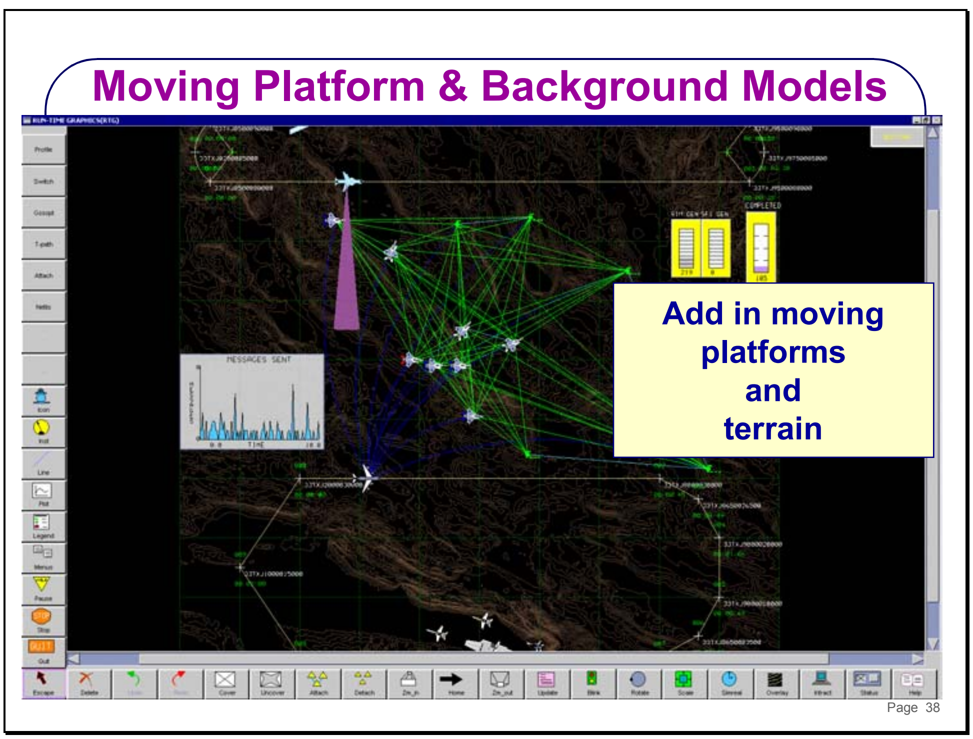
Slide 36



Slide 37



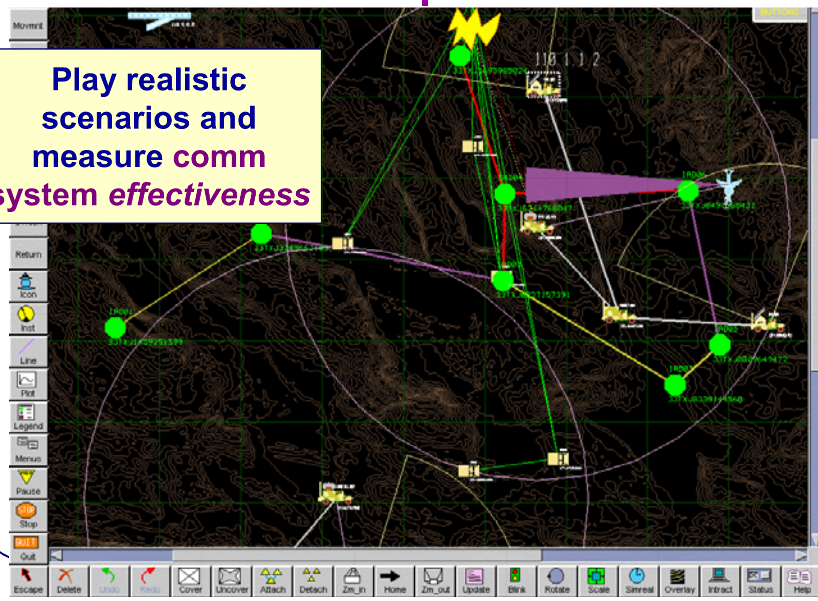
Slide 38



Slide 39

Close the loop with target, sensor, C2 and weapon models

Play realistic scenarios and measure comm system *effectiveness*



Page 39

Slide 40

VISUAL INTERACTIVE SIMULATION:

- Create and save changes / excursions while the simulation is running.
- Modify simulation / scenario graphically by icon / panel manipulation.

Page 40

Slide 41

GIESim

What can we do?:
- in FY03
- in FY04

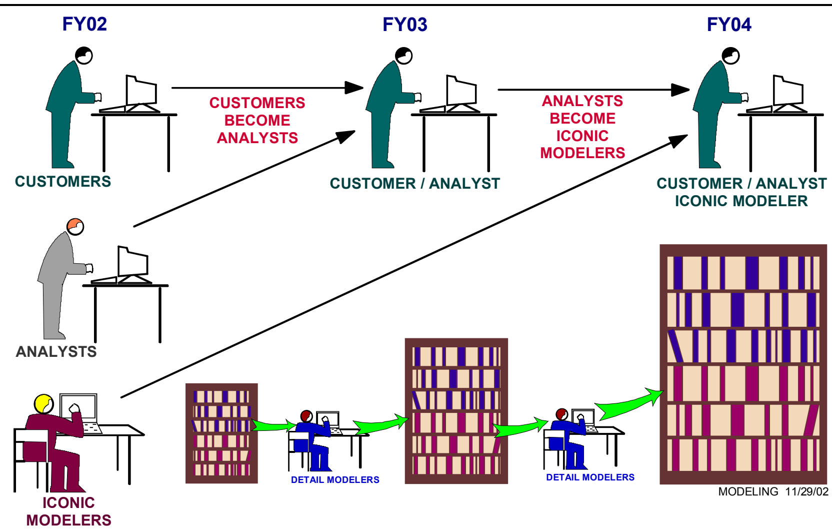
PREDICTION SYSTEMS, INC.

GIESim

Page 41

Slide 42

GIESim - Growing Users & Infrastructure



WE MUST: Bring in Users and Grow the Infrastructure

Page 42

Slide 43

GIESim - Generic Approach

To bring in users and grow the infrastructure, we must:

- **Use existing simulation paradigms**
  - **demonstrate ease of complex analysis**
  - **using people who are *not* analysts**
- **Demonstrate use of complex scenarios**
  - **using interactive graphics**
  - **with people who are *not* modelers**
- **Put new models on the shelf - *fast***
  - **using interactive graphics**
  - **and subject-area experts**

GIESim PREDICTION SYSTEMS, INC.

Page 43

Slide 44

GIESim - Generic Approach

We must get out and demonstrate GIESim capabilities to prospective users, e.g.,:

POTENTIAL USERS:

Air Force - AC2ISR, ESC, AFWIC, NAIC, PACAF, …

DoD - DARPA, OSD/C3I, DISA, USJFCOM, USSPC , …

USN - SPAWAR , …

USA - CECOM (I2WD, C2D, …

Homeland Defense - NorthCOM , USSPC , PACOM, FEMA, NIPC

We can start with those that can use what we have right now!

GIESim PREDICTION SYSTEMS, INC.

Page 44

Slide 45

PREDICTION SYSTEMS, INC.

PREDICTION & CONTROL SYSTEM ENGINEERS

309 Morris Ave
Spring Lake, NJ 07762

(732) 449-6800
Fax: (732) 449-0897

www.predictsys.com
psi@predictsys.com

ATTACHMENT II – GSS ATTRIBUTES SLIDES

Slide 1

GSS

Attributes applicable to

GIESim

GIESim

PREDICTION SYSTEMS, INC.

Page 1

Slide 2

GSS Attributes applicable to *GIESim*

Ease of Use:

- **CAD Interface - Easy to learn and use**
- **Graphically build and control large model libraries**
- **Interact graphically with the simulation - *while it's running***
- **Build and modify networks at run-time - using icon hierarchies**

Development Time / Run Time:

- **Rapid development of large simulations**
- **Optimized for high speed execution (10x-100x faster than competitors)**
- **Built-in support for constrained optimization / worst case design**

Support for multi-computer simulation and parallel processing:

- **HLA, DIS, TCP/IP, Shared Resources, Inter-processor Resources**
- **Built-In Multi-Computer Simulation capability**

GIESim

PREDICTION SYSTEMS, INC.

Page 2

Slide 3

GSS Attributes applicable to *GIESim*

Scalability:

- Support for millions of icons
- Huge event queue (over 32,000 and easily expanded beyond 1M)
- Can handle many thousands of complex entities
- Easy migration to parallel processing environments

Data Interface Capabilities:

- Standard File Interface
- Text and binary files, direct access files, fixed/variable length records
- XML parsers, direct SQL database access (in 11.0)
- TCP/IP Networking
- XML, HTML output (direct PowerPoint)

Platform independence:

- Uses OpenGL and 'C' compiler
- Identical on Linux, Solaris, IRIX, AIX, WINDOWS(NT4.0, 2000, XP)

Slide 4

GSS Attributes applicable to *GIESim*

Configuration Control of models and architectures:

- Enforced by engineering drawings
- Supports hierarchical modeling
- Supports interactive, hierarchical use of icons.

Cost of Ownership:

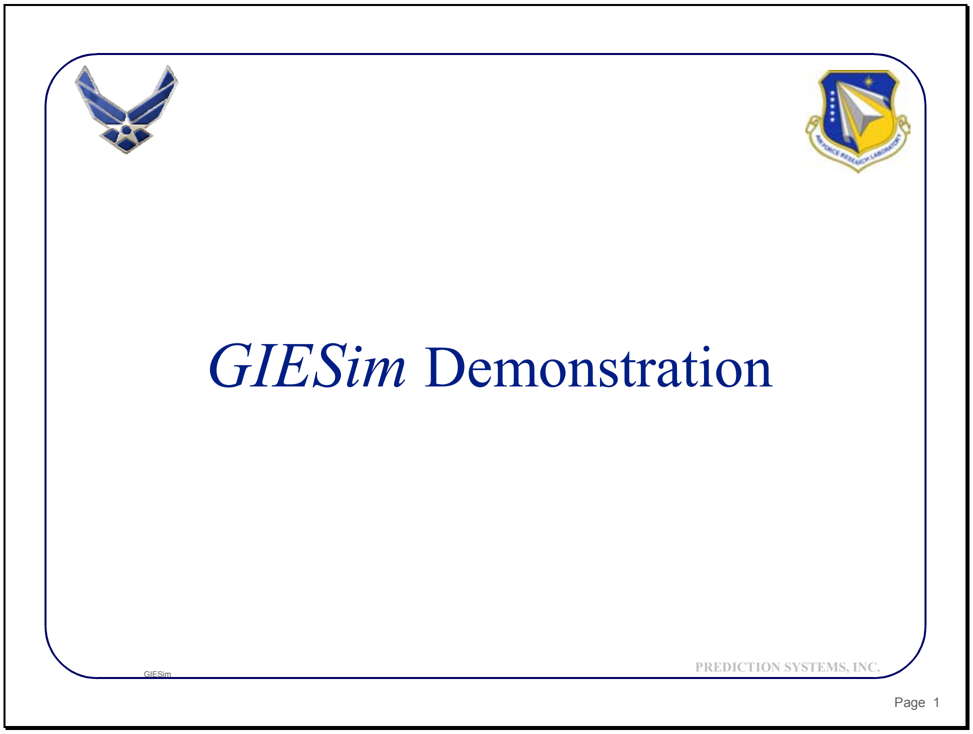
- Low Licensing Fees (free for clients)
- Easy reuse of models and simulations
- Reduced maintenance time and cost (no maintenance fees)

Track Record / Existing Infrastructure:

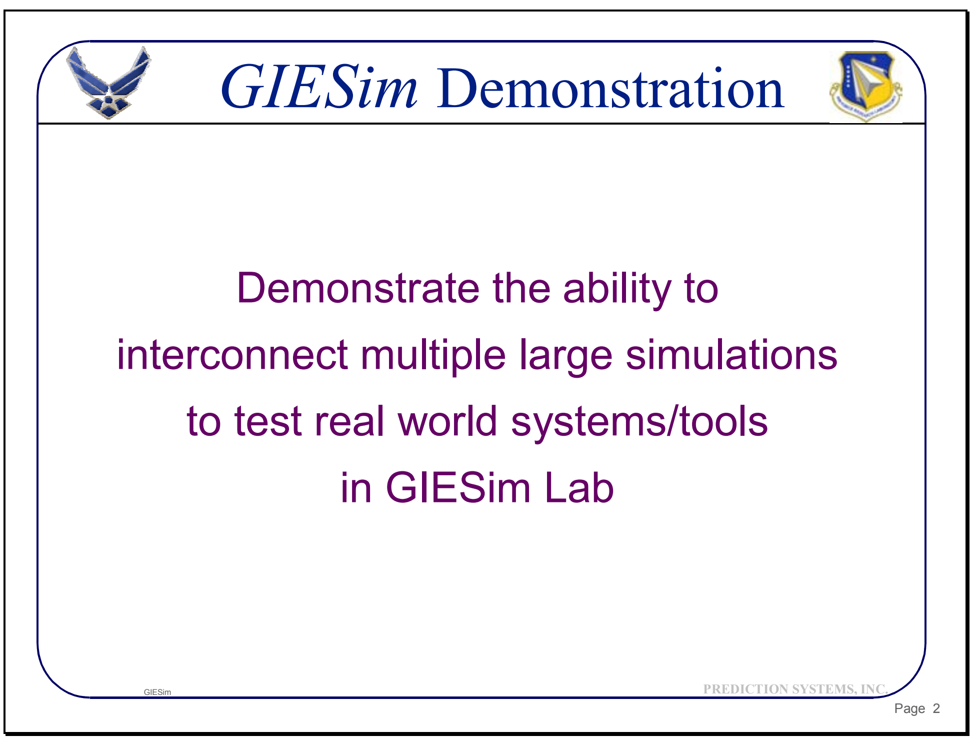
- Satisfied hundreds of DoD client needs for complex simulations and planning tools
- Large collection of models and simulations to draw on
- Large and rapidly growing client reference list

ATTACHMENT III – GSS DEMONSTRATION OVERVIEW SLIDES

Slide 1



Slide 2



Slide 3

GIESim Demonstration

Ground Truth Sim - simulates ground truth position of various emitters.

Sensor Sim - simulates various sensor systems - on the ground and in the air.

Operations Management System (OMS) - used to fuse and manage multiple sensor systems.

EBO-IADS Sim - used to plan missions over the IADS.

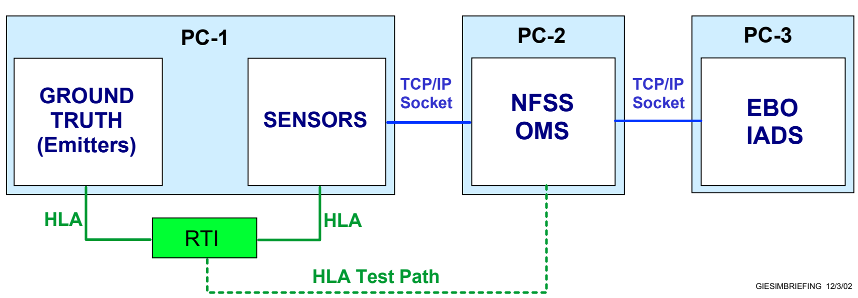
GIESim

PREDICTION SYSTEMS, INC.

Page 3

Slide 4

GIESim Demonstration



GIESim

PREDICTION SYSTEMS, INC.

Page 4

Slide 5

GIESim Demonstration

Ground Truth Sim **- sends data to Sensor Sim via HLA.**

Sensor Sim **sends updates to OMS via TCP/IP Socket.**

OMS **sends fused/updated data to IADS via TCP/IP.**

IADS Sim **sets sensor positions based on OMS data.**

IADS Sim **used to test effectiveness of escort jammer**
- to knockout the IADS.

GIESim PREDICTION SYSTEMS, INC

Page 5